

MAGAZINE

BSD

FOR NOVICE AND ADVANCED USERS

OPENBSD AND THE STATE OF GAMING

HOW TO ADD A NEW SYSTEM TUNABLE TO FREEBSD

QUICKSTART WITH KUBERNETES AND GKE

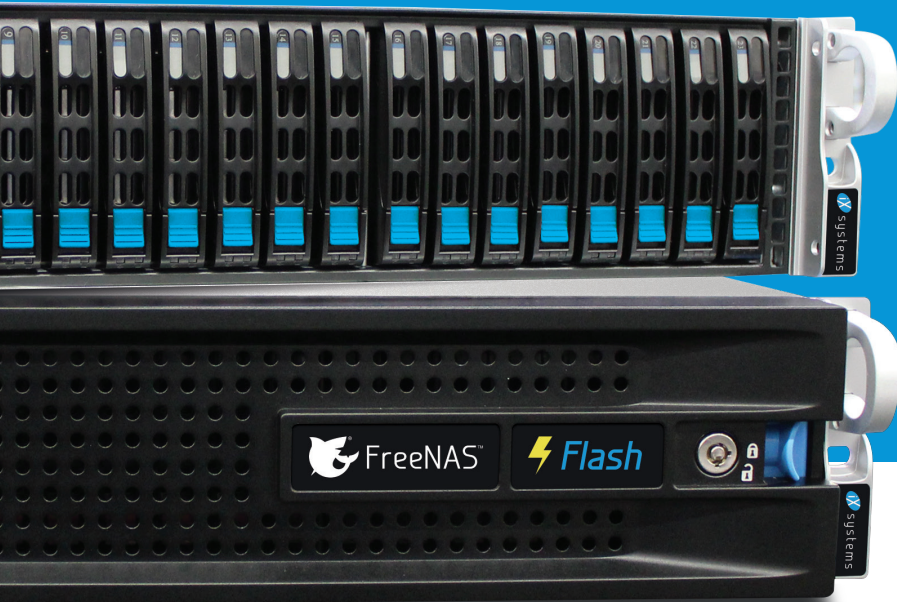
KUBERNETES..! ERA OF INNOVATION

HOW TO MANAGE MULTIPLE PERL 6 INSTALLATIONS
WITH RAKUDOBREW

VOL 12 NO 03

ISSUE 03/2018 (103)

ISSN 1898-9144



IS AFFORDABLE FLASH STORAGE OUT OF REACH?

NOT ANYMORE!

IXSYSTEMS DELIVERS A FLASH ARRAY FOR UNDER \$10,000.

Introducing FreeNAS® Certified Flash: A high performance all-flash array at the cost of spinning disk.

- ⚡ Unifies NAS, SAN, and object storage to support multiple workloads
- ⚡ Runs FreeNAS, the world's #1 software-defined storage solution
- ⚡ Performance-oriented design provides maximum throughput/IOPs and lowest latency
- ⚡ OpenZFS ensures data integrity
- ⚡ Perfectly suited for Virtualization, Databases, Analytics, HPC, and M&E
- ⚡ 10TB of all-flash storage for less than \$10,000
- ⚡ Maximizes ROI via high-density SSD technology and inline data reduction
- ⚡ Scales to 100TB in a 2U form factor

The all-flash datacenter is now within reach. Deploy a FreeNAS Certified Flash array today from iXsystems and take advantage of all the benefits flash delivers.

Call or click today! **1-855-GREP-4-IX (US) | 1-408-943-4100 (Non-US) | www.iXsystems.com/FreeNAS-certified-servers**

DON'T DEPEND ON CONSUMER- GRADE STORAGE.

KEEP YOUR DATA SAFE!



USE AN ENTERPRISE-GRADE STORAGE SYSTEM FROM IXSYSTEMS INSTEAD.

The FreeNAS Mini: Plug it in and boot it up – it just works.

- Runs FreeNAS, the world's #1 software-defined storage solution
- Unifies NAS, SAN, and object storage to support multiple workloads
- Encrypt data at rest or in flight using an 8-Core 2.4GHz Intel® Atom® processor
- OpenZFS ensures data integrity
- A 4-bay or 8-bay desktop storage array that scales to 48TB and packs a wallop
- Backed by a 1 year parts and labor warranty, and supported by the Silicon Valley team that designed and built it
- Perfectly suited for SoHo/SMB workloads like backups, replication, and file sharing
- Lowers storage TCO through its use of enterprise-class hardware, ECC RAM, optional flash, white-glove support, and enterprise hard drives

And really – why would you trust storage from anyone else?



Call or click today! **1-855-GREP-4-IX** (US) | **1-408-943-4100** (Non-US) | www.iXsystems.com/Freenas-Mini or purchase on Amazon.

MAGAZINE BSD

The Editor's Word

Dear Readers,

I hope this finds you well and in a happy mood since the start of Spring. Today, I am pleased to announce the release of the BSD Magazine issue. I hope it will bring lots of joy, happiness, and fulfilment to you. This is also a special time for those who are waiting for Easter celebration like me. I am optimistic that the holiday period brings hope and faith to sustain us in the coming days. Thus, take delight during this period.

Now, let's talk about the issue you have just downloaded. As the norm, you will find a collection of articles. This time, we prepared 8 interesting and informative articles for this issue which are worth your read. The articles were written by experts in various fields to provide you with highest quality knowledge. For this issue, the articles were submitted by Luca Ferrari, Leonardo Neves, Moustafa Nabil El-Zeny, Albert Hui, Carlos Neira, Abdorrahman Homaei, and David Carlier. And for your usual dessert, please see what Rob Somerville has instore for you this time. We also really love his columns and we are eager to see what will be his next submission for next month.

If any question arises in your mind during or after reading the articles, please feel free to contact me. We hope you enjoy reading this issue and develop your new skills with our magazine!

As long as we have our precious readers, we have a purpose. We owe you a huge Thank You. We are grateful for every comment and opinion, either positive or negative. All comments are welcome. Every word from you not only lets us improve the BSD magazine, but also brings us closer to the ideal shape of our publication.

*Thank you and Happy Easter,
Ewa & the BSD team*

TABLE OF CONTENTS

In Brief

In Brief

08

Ewa & The BSD Team

This column presents the latest news coverage of breaking news, events, product releases, and trending topics from the BSD sector.

Perl

How to Manage Multiple Perl 6 Installations with Rakudobrew

10

Luca Ferrari

Perl 6 is a language in the Perl family. It is very feature rich and oriented to several programming paradigms, including the Object Oriented one. Rakudobrew is a tool that helps in installing and managing different installations of a runnable Perl 6 environment, and offers an easy way to get a Perl 6 instance on a machine.

Kubernetes

Quickstart with Kubernetes and GKE (Part 1/2)

14

Leonardo Neves

This article will discuss how to deploy a simple docker application on Google GKE. Readers will be able to deploy any application public available on docker hub on GKE, taking many advantages from that platform, like high availability using several data-centers and unlimited scalability.

Kubernetes..! Era of Innovation

20

Moustafa Nabil El-Zeny

Today, I am going to resume my speech about

OpenShift, K8S, Containers, Orchestrators, etc. When you intend to dive deeper into the Container Orchestration world, you should ask yourself a set of questions... what, which, why and where?

FreeBSD

How to Add a New System Tunable to FreeBSD

24

Carlos Neira

FreeBSD comes out of the box with several system tunable parameters for each of its subsystems. There is a system tunable for virtual memory, file systems, I/O, networking, etc. We will learn how to customize them and also create our own system tunable.

Caddy Web Server On FreeBSD

30

Abdorrahman Homaei

Caddy is an open source, middleware, secure, HTTP/2-enabled web server written in Go programming language that has been created in 2015. Caddy configuration and initiation is simple and clear and lets you create an HTTPS-enabled website in 5 seconds.

OpenBSD

OpenBSD and The State of Gaming

34

David Carlier

OpenBSD is already well-known for its security strengths, but among its third party software, it can also be used to entertain the user.

OVS

Open vSwitch Overview

Albert Hui

Open vSwitch (OVS) is an open source software-defined networking solution to deliver software data center infrastructure as a service functionality for today's cloud-based paradigms. OVS was built and based upon Stanford University's OpenFlow project. OVS functions both as a router and switch. Therefore, it is also referred to as a multilayer switch by examining content from the Open System Interconnection (OSI) reference model encompassing Layers 2 through Layer 7.

Presentation

How to Assist the Business World with OTRS? 42

María Polett Ramos

Column

With the latest chemical attack in the UK that has critically injured two individuals and seriously injured a serving police officer, what are the geopolitical, media, and technical implications of this latest outrage? 50

Rob Somerville

36

MAGAZINE BSD

Editor in Chief:

Ewa Dudzic
ewa@bsdmag.org
www.bsdmag.org

Contributing:

Sanel Zukan, Luca Ferrari, José B. Alós, Carlos Klop, Eduardo Lavaque, Jean-Baptiste Boric, Rafael Santiago, Andrey Ferriyan, Natalia Portillo, E.G Nadhan, Daniel Cialdella Converti, Vitaly Repin, Henrik Nyh, Renan Dias, Rob Somerville, Hubert Feyrer, Kalin Staykov, Manuel Daza, Abdorrahman Homaei, Amit Chugh, Mohamed Farag, Bob Cromwell, David Rodriguez, Carlos Antonio Neira Bustos, Antonio Francesco Gentile, Randy Ramirez, Vishal Lambe, Mikhail Zakharov, Pedro Giffuni, David Carlier, Albert Hui, Marcus Shmitt, Aryeh Friedman

Top Betatesters & Proofreaders:

Daniel Cialdella Converti, Eric De La Cruz Lugo, Daniel LaFlamme, Steven Wierckx, Denise Ebery, Eric Geissinger, Luca Ferrari, Imad Soltani, Olaoluwa Omokanwaye, Radjis Mahangoe, Katherine Dizon, Natalie Fahey, and Mark VonFange.

Special Thanks:

Denise Ebery
Katherine Dizon

Senior Consultant/Publisher:

Paweł Marciniak

**Publisher: Hakin9 Media SK,
02-676 Warsaw, Poland Postepu 17D, Poland
worldwide publishing
editors@bsdmag.org**

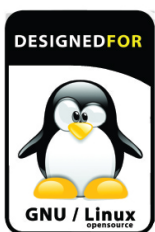
Hakin9 Media SK is looking for partners from all over the world. If you are interested in cooperation with us, please contact us via e-mail: editors@bsdmag.org

All trademarks presented in the magazine were used only for informative purposes. All rights to trademarks presented in the magazine are reserved by the companies which own them.



Rack-mount networking server

Designed for BSD and Linux Systems



Designed. Certified. Supported

Up to **5.5Gbit/s**
routing power!



KEY FEATURES

- ▶ 6 NICs w/ Intel igb(4) driver w/ bypass
- ▶ Hand-picked server chipsets
- ▶ Netmap Ready (FreeBSD & pfSense)
- ▶ Up to 14 Gigabit expansion ports
- ▶ Up to 4x10GbE SFP+ expansion



PERFECT FOR

- ▶ BGP & OSPF routing
- ▶ Firewall & UTM Security Appliances
- ▶ Intrusion Detection & WAF
- ▶ CDN & Web Cache / Proxy
- ▶ E-mail Server & SMTP Filtering

How To Install Apache, MariaDB & PHP (FBAMP) on FreeBSD



Augusto Dueñas posted a very useful tutorial on how to install some useful tools and application on the FreeBSD system. He explained why and what we need to do in this process to have a complete and functional system. *“One of these operating systems is **FreeBSD** which is a derivative of BSD, the UNIX version for compatible x86 architectures. In this opportunity, we will see how we can install FBAMP, or as we know in some versions of Linux as LAMP in this FreeBSD system”*

Source:

<https://thelinuxcode.com/install-apache-mariadb-php-fbamp-freebsd/>

Open-Source Summit Europe 2018 Call for Proposals



October 22-24, 2018, Edinburgh, Scotland, UK
The call for proposals for the 2018 Open-Source Summit Europe is now open. The Open-Source Summit Europe will be held October 22-24, 2018, in Edinburgh, Scotland, UK. More information and a list of suggested topics can be

found here. We're hoping to get a few FreeBSD talks into this traditionally Linux-focused event. If you have an idea for a presentation that will fit into one of the suggested categories but you aren't sure how to proceed, please contact us.

Source:

https://linuxfoundation.smapply.io/prog/open_source_summit_europe_2018/

Looking at Lumina Desktop 2.0

Ken Moore, Lead Developer of the TrueOS Project, answered some of the most frequently asked questions about Lumina Desktop from the open-source community. All was gathered by John Smith.

“Ken: Lumina Desktop 2.0 is a significant overhaul compared to Lumina 1.x. Almost every single subsystem of the desktop has been streamlined, resulting in a nearly-total conversion in many important areas.

With Lumina Desktop 2.0, we will finally achieve our long-term goal of turning Lumina into a complete, end-to-end management system for the graphical session and removing all the current runtime dependencies from Lumina 1.x (Fluxbox, xscreensaver, compton/xcompmgr).

The functionality from those utilities is now provided by Lumina Desktop itself.

Going along with the session management changes, we have compressed the entire desktop into a single, multi-threaded binary. This means that if any rogue script or tool starts trying to muck about with the memory used by the desktop (probably even more relevant now than when we started working on this), the entire desktop session will close/crash rather than allowing targeted application crashes to bypass the session security mechanisms. By the same

token, this also prevents “man-in-the-middle” type of attacks because the desktop does not use any sort of external messaging system to communicate (looking at you `dbus`). This also gives a large performance boost to the Lumina Desktop

The entire system for how a user’s settings get saved and loaded has been completely redone, making it a “layered” settings system which allows the default settings (Lumina) to get transparently replaced by system settings (OS/Distributor/SysAdmin) which can get replaced by individual user settings. This results in the actual changes in the user setting files to be kept to a minimum and allows for a smooth transition between updates to the OS or Desktop. This also provides the ability to “restrict” a user’s desktop session (based on a system config file) to the default system settings and read-only user sessions for certain business applications.

The entire graphical interface has been written in QML in order to fully-utilize hardware-based GPU acceleration with OpenGL while the backend logic and management systems are still written entirely in C++. This results in blazing fast performance on the backend systems (myriad multi-threaded C++ objects) as well as a smooth and responsive graphical interface with all the bells and whistles (drag and drop, compositing, shading, etc).”

Source:

<https://www.trueos.org/blog/looking-lumina-desktop-2-0/>

ZFS User Conference

It is a great event where you can listen to one of the founders of ZFS talk about ZFS’s history and future. You will learn how to be more effective at administering ZFS environments with intermediate ZFS training and hear about interesting ZFS use cases. Finally, you learn about exciting new improvements and developments in ZFS.

Date: 19 Apr 2018 to 20 Apr 2018

Location: Norwalk, CT, USA

Source: <http://zfs.datto.com/index.html>

NetBSD 7.1.2 Released



The NetBSD Project is pleased to announce NetBSD 7.1.2, the second security/critical update of the NetBSD 7.1 release branch. It represents a selected subset of fixes deemed important for security or stability reasons. Complete source and binaries for NetBSD 7.1.2 are available for download at many sites around the world. A list of download sites providing FTP, AnonCVS, and other services may be found at <https://www.NetBSD.org/mirrors/>. We encourage users who wish to install via ISO or USB disk images to download via BitTorrent by using the torrent files supplied in the images area. A list of hashes for the NetBSD 7.1.2 distribution has been signed with the well-connected PGP key for the NetBSD Security Officer: https://ftp.NetBSD.org/pub/NetBSD/security/hashes/NetBSD-7.1.2_hashes.asc

Source:

<https://www.netbsd.org/releases/formal-7/NetBSD-7.1.2.html>

How to Manage Multiple Perl 6 Installations with Rakudobrew

Perl 6 is a language in the *Perl family*. It is very feature rich and oriented towards several programming paradigms, including the Object Oriented programming. Rakudobrew is a tool that helps with installing and managing different installations of a runnable Perl 6 environment, and offers an easy way to get a Perl 6 instance on a machine.

What you need to know

- Basic Perl knowledge and terminology
- Basic FreeBSD shell knowledge

What you will learn

- How to install rakudobrew, initialize and run it
- How to install different Perl 6 interpreters on the same machine, and how to use a specific one depending on your needs
- How to manage Perl 6 interpreters

Introduction

Perl 6 is a quite a young language in the *Perl family*, and therefore it is often not installed on many systems by default as opposed to its younger cousin Perl 5.

Rakudobrew is a Perl program that allows users to download, build, and run Perl 6 instances in their own space, without having to affect the system-wide installation (if installed) of Perl 6 or to have administrative privileges. The philosophy is similar to other *brew* suites.

Perl 6 is a complex beast when compared to Perl 5 because it requires a *virtual machine* to run, has a separate package manager and requires specific compilation. Rakudobrew simplifies the steps required to get all the pieces up and running - downloading, compiling and installing every necessary part.

In Perl 6 terminology, it is important to distinguish the following:

Rakudo a Perl 6 compiler;

Rakudo-star a Perl 6 compiler with several modules included;

backend a virtual machine able to run any piece of Perl 6 code compiled by a compiler;

nqp (Not Quite Perl) a Perl-like language used to drive low-level virtual machine operations;

perl6 the effective (and interactive) implementation of a Perl 6 executable.

From the above, to allow a Perl 6 source code to run, it is necessary that the source code is compiled on the fly by a compiler and is executed by a virtual machine.

Rakudobrew was primarily born to allow Perl 6 developers and testers to install and run different Perl 6 environments in an easy way. Additionally, it had been adopted in the past as a way of installing Perl 6 for regular users too. It is worth noting that, by design, rakudobrew downloads and compiles a *tagged* version of the Perl 6 source code that may not necessarily be the optimal or most stable one available at the moment. Therefore, before using rakudobrew yourself, keep in mind that, while powerful, it might not be the recommended tool to adopt. Hence, the aim of this paper is just to present it as a short and sweet way to get a *recent* version of Perl 6 up and running. But for production environments, official Perl 6 releases should be preferred. Official *Rakudo* and *Rakudo-star*

releases can be downloaded for several platforms from the official website.

Installing rakudobrew

Rakudobrew is neither available in ports nor in packages, hence the only way to install it is from source. Since the repository is kept under a *GitHub*, git and an internet connection are required to download it.

As a normal user, simply provide the following command to download:

```
% git clone
https://github.com/tadzik/rakudobrew
~/.rakudobrew
```

The repository will be cloned into the hidden.rakudobrew folder under your home folder. Of course, it is possible to move it to another location. In this article, the default installation path, \$HOME/.rakudobrew , will be assumed.

Once rakudobrew has been downloaded, it must be *initialized* to work properly. First of all, let's check that the executable is working:

```
% ~/.rakudobrew/bin/rakudobrew
Usage:
rakudobrew current
rakudobrew list-available
rakudobrew build
rakudobrew build zef
...
```

It is worth noting that the executable of rakudobrew is a Perl 5 script, meaning the system must have a working version of Perl 5 to use it. In case a specific version of Perl 5 is required, please refer to the previous article on *Managing Multiple Perl 6 Installations with Perlrew* in the magazine issue 2018-01.

Once the rakudobrew executable is running, it is possible to configure it for permanent usage with the init command. The init command will produce a shell function and set a few

environment variables to allow the user to use the rakudobrew executable; such shell configuration has to be included into the shell configuration files (profile or rc files).

```
% ~/.rakudobrew/bin/rakudobrew init - >>
~/.zprofile
```

After the shell has been configured to use rakudobrew, it is possible to open a new shell or logout/login (depending on the type of shell and its configuration) to see the changes. If everything worked fine, the rakudobrew executable can be launched without the path specification.

The rakudobrew executable works on a *command-oriented* interface: each action is specified by a particular command that can optionally take arguments. Therefore, a command must be specified to make rakudobrew do something.

Installing Perl 6

Once rakudobrew is working, it is possible to install a new Perl 6 executable. First of all, it is possible to ensure nothing is in use:

```
% rakudobrew current
Not running anything at the moment. Use
'rakudobrew switch' to set a version
```

```
% rakudobrew switch
Switch to what?
Available builds
```

As readers can see, rakudobrew complains about the fact that no Perl 6 executable is currently enabled, and that it is not possible to switch to any version since the Available builds is empty.

To install a new Perl 6 environment it is required to build it. The build command asks for a Perl 6 version, as well as backend engine.

Perl 6 versions are numbered monthly, so for instance 2017.12 is the *december 2017* release. The backend engine is the virtual machine that will execute Perl 6 – currently the *Java Virtual Machine* and *MoarVM* are supported, with the latter being the *official* Perl 6 virtual machine.

Having stated the above, it is possible to search for an instance to build with the list-available command, and then use the build one to compile the instance.

```
% rakudobrew list-available
Available Rakudo versions:

...
2017.11
2017.12
2018.01
v6.b
v6.c
```

```
Available backends:

jvm
moar
moar-blead
```

```
% rakudobrew build moar 2018.01
...
```

The build command can take a while, depending on the available computer resources.

After the build has completed, the new version of Perl 6 is listed through the list command. For instance after having built a few instances, the situation could be as follows:

```
% rakudobrew list
jvm-2017.09
moar-2016.12
moar-2017.09
moar-2017.11
moar-2017.12
* moar-2018.01
moar-blead-2017.11
```


The entry with a leading asterisk is the *current* running instance, also reported by the current command:

```
% rakudobrew current
Currently running moar-2018.01
```

In order to select which Perl 6 environment to use, the switch command is used: it is necessary to specify which instance to switch to, and rakudobrew will update the environment:

```
% rakudobrew switch moar-2017.12
Switching to moar-2017.12
```

```
% rakudobrew current
Currently running moar-2017.12
```

Installing modules

Perl 6 uses the *Zef* module installer to install modules. To some extent, *Zef* is the counterpart of the `cpan` and `cpanm` commands for Perl 5.

The *Zef* module installer has to be built through rakudobrew, and the `build zef` command does exactly that:

```
% rakudobrew build zef
```

For every instance of Perl 6, *Zef* has to be built, otherwise it will not be usable on the *current* running environment. Once *zef* is installed, it is possible to run it with the `install` command and a module name. For instance:

```
% zef install Archive::SimpleZip
==> Searching for: Archive::SimpleZip
...
==> Installing:
Archive::SimpleZip:ver<0.1.2>
```

In order to see every *zef* command and available options, just run the command without any argument.

Conclusions

Rakudobrew is a powerful tool in the *brew* family that allows for quick and easy installation of a

Perl 6 environment without requiring administrative privileges or tainting system-wide installation (if any).

Moreover, with rakudobrew, it is possible to manage and run different instances and versions of Perl 6 thus allowing users to experiment with features and portability.

References

Perl 6 official website: <https://perl6.org/>

Rakudo (and Rakudo-start) official website: <http://rakudo.org>

Rakudobrew GitHub repository: <https://github.com/tadzik/rakudobrew>

MoarVM official website: <https://www.moarvm.org/>

Perl 6 modules directory: <https://modules.perl6.org/>

Meet the Author

Luca Ferrari lives in Italy with his beautiful wife, his great son, and two female cats. Computer science passionate since the Commodore 64 age, he holds a Master's Degree and Ph.D. in Computer Science. He is a PostgreSQL enthusiast, a Perl lover, an Operating System enthusiast, a UNIX fan and performs as many tasks as possible within Emacs. He considers Open-Source the only sane way of creating software and services. His website is available at <http://fluca1978.github.io>

Kubernetes

Quickstart with Kubernetes and GKE (Part 1/2)

This article will discuss how to deploy a simple Docker application on Google Kubernetes Engine (GKE). Readers will be able to deploy any application publicly available on Docker Hub on GKE, benefiting from many advantages that platform provides - like high availability using several data-centers and scalability.

What you will learn...

- How to get started with Kubernetes quickly
- How to get started with GKE
- How to deploy a simple Docker application on GKE

What you should have...

- basic understanding of Linux and Linux commands
- basic understanding of Docker

Introduction

Docker is relatively new, but it's already widely used and is quickly taking over data-centers all over the world. Initially used just by developers, it's now being adopted by all kind of companies at a remarkable rate.

Kubernetes enhances Docker in virtually all missing capabilities. It takes care of important parts of the environment like management, high availability, self-healing, scaling and optimizes automated deployment.

Using just Docker and Kubernetes, you already have a very robust environment, probably much more reliable than using traditional technologies

like virtual or physical machines, load balancers and configuration managers. But we still can improve the environment using a cloud provider. With a public or private cloud provider we will have management, high availability, self-healing, scaling also in the bottom layer, where an operating system runs and hosts the Kubernetes service. The cloud provider that supports Kubernetes natively is GKE (Google Kubernetes Engine) from Google and it will be used in this article.

Getting used to new technologies takes time. You can learn through books, tutorials, courses, etc. but to master the technology there is nothing better than hands-on experience. In this article you will learn how to start using Docker, Kubernetes and GKE quickly. Having your new environment ready, it will be easy to play around and learn more about all the technologies.

The many advantages of using Docker, Kubernetes and GKE

Why Docker?

There are several advantages of using Docker rather than virtual machines or physical machines. First, Docker reduces the infrastructure resources needed to run an application. Second, Docker helps with portability - you can move your application to different platforms easily. Third, it will boost your deployment process since Docker fits better in and agile environment with CI/CD techniques. Last but not least, Docker can help you isolate applications properly, making your environment much more secure.

How about the production environment?

Docker was not initially developed to work in production environments, where features like high-availability and scaling are very important. Despite that, just after the first versions of Docker were launched many companies started developing or integrating existing cluster services

to support Docker. The most significant cluster technologies that support Docker natively are Docker Swarm, Apache Mesos and Google Kubernetes.

Why Kubernetes?

Kubernetes, also known as k8s, is the most advanced system that orchestrates containers. Originally created by Google it is now an open-source software maintained by Cloud Native Computing Foundation. Kubernetes manages automating deployment, scaling, high-availability. You could say Kubernetes is like a cluster on steroids.

Kubernetes is state oriented

When properly configured, Kubernetes will keep a desired state, that is, it will make sure all the requested pods/containers, load balancers, services and so on are running. When we demand a state change, Kubernetes will do everything that's needed without disrupting the services. The same will happen in case of hardware issues or issues in the operating system that host the Kubernetes environment.

Getting more advantages using cloud providers

Even when using Kubernetes and getting all advantages that it offers, we will still need an environment to host it. Even though we can install Kubernetes directly on operating systems we have a lot of other benefits if we use a cloud environment. Using a cloud environment, the provider will manage the operating system for you and you don't need to be concerned about patches and optimizations. The provider can also scale out when more hosts are needed and remove hosts when the demand decreases. Another big advantage of using a cloud provider is that they have multiples data-centers spread in the same zone, with redundant links and redundant power supplies, the perfect environment to run a Kubernetes environment.

GKE is currently the best cloud provider for Kubernetes

We have many cloud providers available in the market, most of them offer a very good level of service, however Google Kubernetes Engine, or GKE, is currently the most advanced of them. Google created Kubernetes and they have been working on optimizations on Kubernetes and GKE ever since. Another important consideration is that Google also uses GKE to host their most critical services, it's like a warranty that the service has a very good level of quality.

GKE is very easy to use

GKE is also very simple to use and you can launch a Docker application there in a matter of minutes. The most amazing thing is that your Docker/Kubernetes/GKE Environment will have a level of availability similar to critical services of big companies. And your environment, even though small in the beginning can grow to thousands of Docker containers and hosts without any disruption.

Creating the GCP account

GKE is part of Google Cloud Platform, or GCP. You will run Kubernetes on top of some GCP hosts, as you will see.

To proceed with the sign in, go to the link <https://cloud.google.com/kubernetes-engine/>

and hit the button 'TRY IT FREE', as you can see on Figure 1:



Figure 1: Kubernetes Engine - Try it Free

After that you will just need to accept all the terms to continue to the next step. Next, you will need to create a Payment Profile. It requires you

to enter your credit card information. When joining GCP you have 12 months trial to use U\$300 in credit, it's sufficient to create a small environment with a Kubernetes cluster. Even if you create a lot of resources inside GCP and spend your U\$300 credit too fast, Google will notify you when the credits are running out. You will have to pay only if you confirm that after Google send you a message, so don't worry about uninvited bills. As you can see on Figure 2, this payment profile will also be used on all Google products:

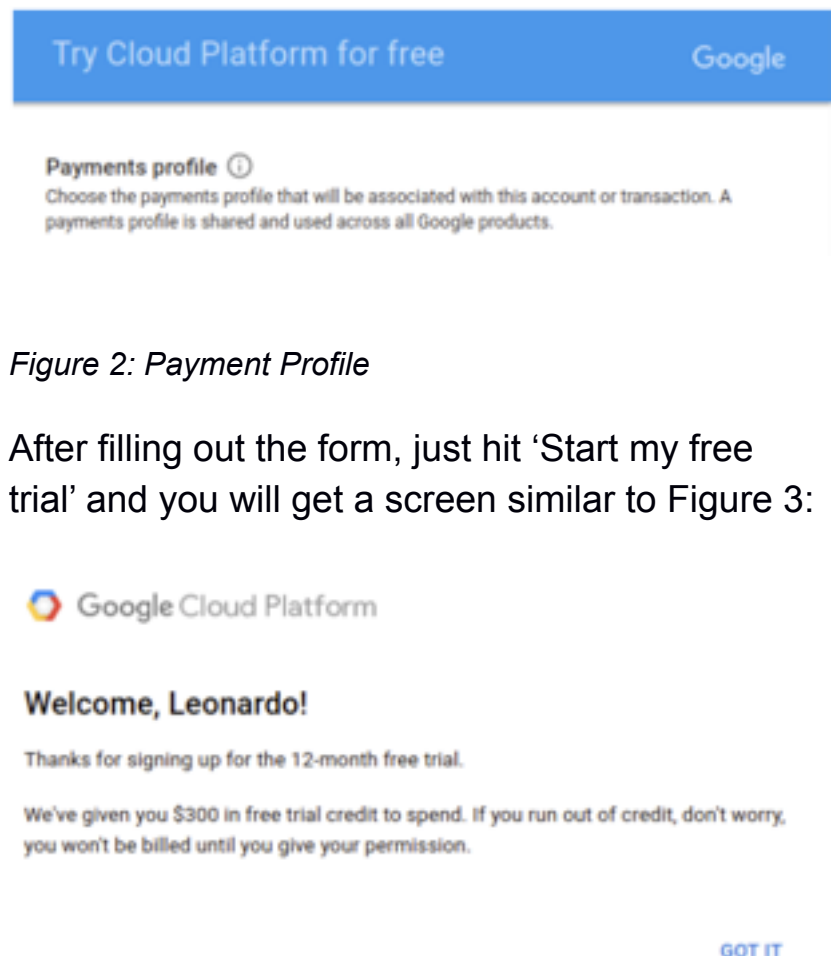


Figure 2: Payment Profile

After filling out the form, just hit 'Start my free trial' and you will get a screen similar to Figure 3:

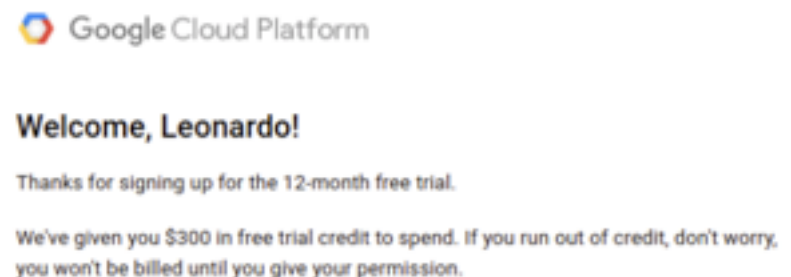


Figure 3: Welcome GCP

As you can see the process is very simple. Now you have a GCP account and you can spin up virtual machines, create disks, images and users and so on. In the next section you will see how to create a project, that's needed to be create before creating any Kubernetes Cluster.

Creating a new project

On GCP, you can create and use multiple projects. Projects allow you to segregate

resources and responsibilities. You can create a project just for developers to test new resources without giving them access to the production project and environment for example. Different projects are on isolated networks, even if they use the same IP ranges. Please notice that projects are different from Kubernetes namespaces. Using namespaces, Kubernetes can isolate a set of containers and its resources from containers and resources from other namespaces, but in this case the hosts running Kubernetes will be the same. In addition to using namespaces, there will be isolation at the application level - there is a possibility that a namespace affects the performance of other namespaces, for instance when the load is too high. The choice between creating different projects or just different namespaces depends on the company, environment and even the type of data that the environment will host. The intention of this article is to get a quick start using the technology therefore complex environments with multiple projects or namespaces are out of scope of this article.

To create a new project on GCP, go to <https://console.cloud.google.com/cloud-resource-manager> and hit 'CREATE PROJECT'. Choose a name and click 'Create'. In case the new project is not showed, go to <https://console.cloud.google.com/cloud-resource-manager> again. Click at the name of the new project, GCP will send you to the dashboard of the project.


Creating the Kubernetes Cluster inside GCP

Now that you have the GCP account and the project, it's time to create the Kubernetes cluster.

Go to <https://console.cloud.google.com/kubernetes/config> and hit 'Create Cluster'. Fill in the information on the form similar to what is showed in Figure 4. Make sure you select '1' for the Size field. By default, GKE will create 3 hosts per zone, so if you run you cluster using three zones it will create 9 hosts. To create the small environment to play around with k8s, you need just 2 hosts

running in different zones. With this environment it's possible to simulate most of the issues faced by a cluster in a production environment. We can simulate what happens when a host crashes, for instance.

A Kubernetes cluster is a managed group of uniform VM instances for running Kubernetes. [Learn more](#)

Name 

myfirstcluster


Description (Optional)

myfirstcluster


Location 

☒ Zonal

☐ Regional (beta)

Zone 

us-central1-a

Cluster Version 


1.8.7-gke.1 (default)

Machine type

Customise to select cores, memory and GPUs.

1 vCPU 3.75 GB memory [Customise](#)

[Upgrade your account](#) to create instances with up to 96 cores

Node image 

'cos' provides better security and performance but has limitations that may affect some users. Use 'Ubuntu' if you are affected by these limitations. Note that Ubuntu requires Kubernetes 1.6.4 or greater. [Learn more](#)

Container-Optimised OS (cos)

Size 

1

Figure 4: Creating a Kubernetes Cluster

Clicking in 'More', you can pick additional zones to run the Kubernetes hosts. In the example, us-central1-a will be the primary zone and us-central1-b will be selected to host the second host. Theoretically, outages will happen only if both us-central1-a and us-central1-b become unavailable, what's many times more unlikely to happen than a single zone crash. Important to note that although us-central1-a and us-central1-b are different physical datacenters, they are still located in the same city or metropolitan area. In Figure 5 you can see how

to add additional zones to your Kubernetes cluster.

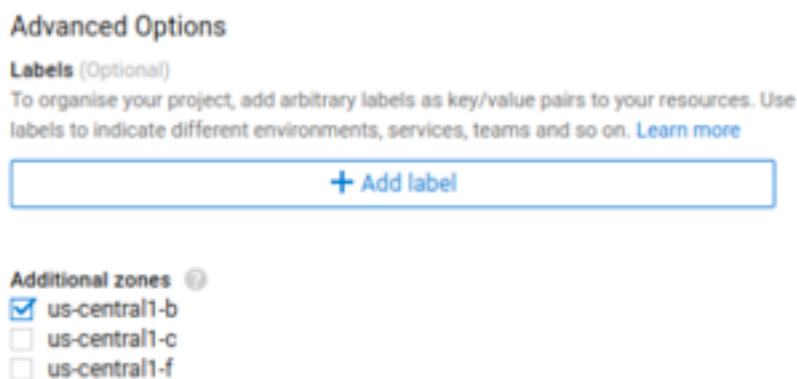


Figure 5: How to add additional zones when creating a Kubernetes cluster

More advanced options

There are a lot of other options that you can test, like the k8s version or auto-updates. Leaving the default options will create an environment sufficient for learning more about Kubernetes, GKE and even Docker. The most amazing thing related to Kubernetes and GKE is that even though this small cluster was created in just a few minutes, it has a very good high-availability level. What once took months and many thousands of dollars to create using physical servers and appliances, can now be done with just a few clicks and dozens of dollars per month. To keep the availability, GKE can also monitor the hosts resources and create new hosts on-demand. When a data-center is unavailable, Kubernetes will start new containers in the good data-center to keep the environment as desired. As you can see, we will have high-availability in two different levels, GKE and Kubernetes.

Unlimited scalability

Another important thing to considerate is the unlimited scalability. You can grow your environment automatically or manually if your small application suddenly become a big

success, with just a few clicks you can grow your environment to the required size. The same can be done in reverse, in case you need to scale down the environment. You will always pay per use and if some cloud provider offers you more advantages compared to GKE, you can simply migrate your environment to it. Kubernetes support is now becoming a de facto standard on cloud providers and migrating a Docker/Kubernetes environment is orders of magnitude easier than migrating traditional services.

How to manage GKE and Kubernetes

Both Kubernetes and GKE were created by Google, so they share many characteristics. For instance, both have a web dashboard, a command line tool and a yaml configuration file (.yaml). You will be surprised how similar they look and this is another point to considerate GKE over other cloud providers. Another important characteristic of both GKE and Kubernetes is that you can full manage the environment from any interface you prefer, in other words, everything you can do through one interface you will be able to do using another interface.

The command line tools, named gcloud (to manage GKE) and kubectl (to manage Kubernetes), can be installed on your desktop or wherever you want. GKE also provides a console with these commands already installed in its web interface, which is very practical.

Accessing the Kubernetes Cluster

At the top right of the page there is a button with a '>' caption (it will show 'Activate Google Cloud Shell' when you hover the mouse over it. Click on it and a console will open on the bottom of the page. Using this console, you can even create a new k8s cluster - as said before you can full manage the resources from any interface.

To manage your recently created k8s cluster, click on the button 'Connect', as you can see on Figure 6:



Name	Location	Cluster size	Total cores	Total memory	Notifications	Labels
myfirstcluster	us-central1-a	2	2 vCPUs	7.50 GB		

Figure 6: Kubernetes Cluster Example

Next, click on 'Run in Cloud Shell' and a gcloud command will be showed. This command will properly configure the kubectl command to manage your cluster. Just hit enter and you will get access to the shell. Now you are able to type any valid gcloud or kubectl command and fully manage both GKE and Kubernetes. To see how powerful kubernetes can be, type 'kubectl config view | more' in the Cloud Shell. A yaml file describing your entire Kubernetes cluster will be showed. You can, for instance, save the output in a file, make some changes and reapply the new file. Yaml files are usually the preferred way to manage Kubernetes clusters.

Conclusions and what's next

As you could see on this article, using GKE is the way to create a Kubernetes cluster with high-availability and unlimited scalability. In this first part we learned how to create the GCP user, the project and the Kubernetes cluster and were introduced to using the Cloud Shell and checking if everything is okay using kubectl config view.

In the second part of the article you will learn more about Kubernetes concepts and find out how to deploy a simple application on it. Using both parts of this article you will be able to launch any application available on Docker Hub using Kubernetes and GKE. Although supporting a Kubernetes production environment will require more learning and practice, creating this small environment is a very good first step to achieve this. You can learn a lot practicing in your personal environment and the U\$300 credit from

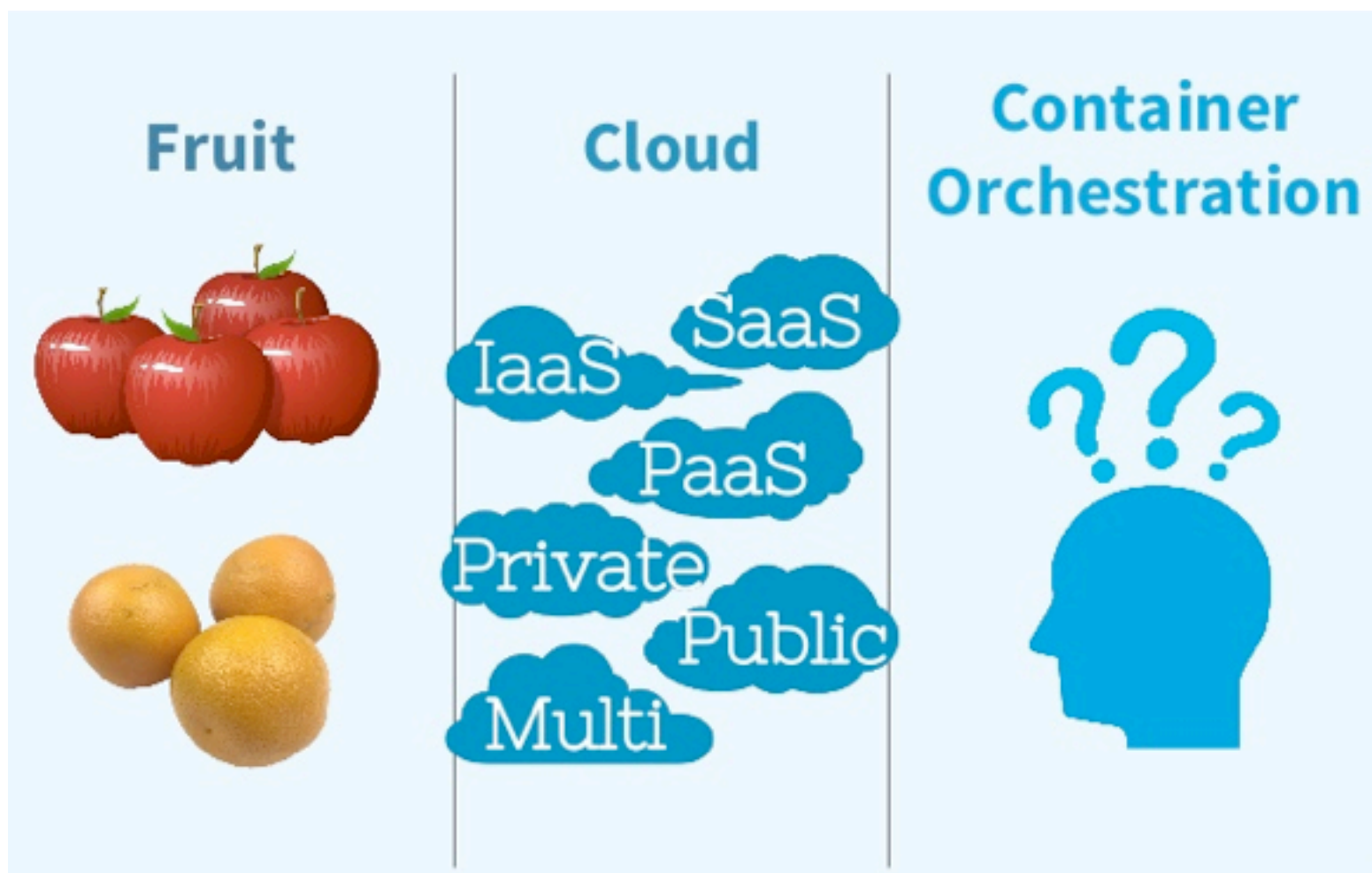
Google allows you to play around for many months. This first part of the article was more theoretical, but still essential. Look forward to the next part, with lots of hands-on material, which is what we geeks really enjoy.

Meet the Author

Leonardo Neves Bernardo got started with Unix in 1996 and since then he is always working with some related technology, in special using Linux systems. He holds many certifications including LPIC-3, LPIC-300, LPIC-302 and LPIC-303, RHCSA and the ITILv3 Foundation. He is from Florianópolis, Brazil, but currently lives in Toronto, Canada, where he is the Security Admin of VerticalScope Inc. His linkedin profile is <https://www.linkedin.com/in/leonardoneves>

Kubernetes

Kubernetes...! An Era of Innovation



Today, I am going to start my series of articles which focus on OpenShift, K8S, Containers, Orchestrators, etc. When you intend to dive deeper into the Container Orchestration world, you should ask yourself a set of questions - What, Which, Why and Where?

✓ What are Container Orchestrators?

These are tools which group hosts to form a cluster. In Development environments, you can get a way with running containers on a single host for testing purposes. However, in Production, you do not have the same liberty.

In addition, you need to ensure that your applications are fault tolerant, scalable, support update/rollback without any downtime, and are accessible from the external world.

✓ Which type of Container Orchestrators do you need?

1- Docker Swarm: Docker Swarm provided by Docker, Inc. It is part of Docker Engine.

2- Kubernetes: K8S was started by Google, but is now a part of the Cloud Native Computing Foundation project.

3- Mesos Marathon: Marathon is one of several frameworks to run containers at scale on Apache Mesos.

4- Amazon ECS: Amazon EC2 Container Service (ECS) is a hosted service provided by Amazon Web Services (AWS).

5- Hashicorp Nomad: Nomad provided by HashiCorp.

✓ Why use Container Orchestrators?



We can argue that containers at scale can be maintained manually, or with the help of some scripts, and can bring multiple hosts together and make them part of a cluster, schedule containers to run on different hosts, help containers running on one host reach out to containers running on other hosts in the cluster, bind containers and storage, keep resource usage in-check, and optimize it when necessary, and allow secure access to applications running inside containers.

✓ Where to deploy Container Orchestrators?

Most container orchestrators can be deployed on the infrastructure of our choice. We can deploy them on bare-metal, VMs, on-premise, or on a cloud of our choice. Also, Kubernetes can be deployed on on a laptop/workstation, inside a company's datacenter, on AWS, on OpenStack, etc. There are even one-click installers available to setup Kubernetes on the Cloud, like Google Container Engine on Google Cloud, or Azure Container Service on Microsoft Azure.

Let's specify one of them and dive deeper into it, in more detail - Kubernetes!

✓ What is Kubernetes ?

"Kubernetes is an open-source system for automating deployment, scaling, and management of containerized applications."

Kubernetes comes from the Greek word **κυβερνήτης**, which means *helmsman* or *ship pilot*. With this analogy in mind, we can think of Kubernetes as the manager for shipping containers.

Kubernetes is also referred to as **k8s**, as there are 8 characters between *k* and *s*.

Kubernetes is highly inspired by the Google Borg system, which we will explore in this chapter. It is an open-source project written in the Go language and licensed under the [Apache License Version 2.0](#).

Kubernetes was started by Google and, with its v1.0 release in July 2015, donated to the [Cloud Native Computing Foundation](#) (CNCF). We will discuss more about CNCF a little later.

Generally, Kubernetes has new releases every three months. The current stable version is 1.7 (as of June 2017).

✓ Kubernetes Features:

Kubernetes offers a very rich set of features for container orchestration. Some of its fully supported features are:

- **Automatic binpacking**
Kubernetes automatically schedules the containers based on resource usage and constraints without sacrificing availability.
- **Self-healing**
Kubernetes automatically replaces and reschedules containers from failed nodes. It also kills and restarts containers which do not respond to health checks based on existing rules and policies..
- **Horizontal scaling**
Kubernetes can automatically scale applications based on resource usage like CPU and memory. In some cases, it also supports dynamic scaling based on customer metrics.
- **Service discovery and load balancing**
Kubernetes groups sets of containers and refers to them via a DNS name. This DNS name is also called a Kubernetes **service**. Kubernetes can discover these services automatically, and load-balance requests between containers of a given service.
- **Automated rollouts and rollbacks**
Kubernetes can roll out and roll back new versions or configurations of an application without introducing any downtime.
- **Secrets and configuration management**
Kubernetes can manage secrets and configuration details for an application without rebuilding the respective images. With secrets, we can share confidential information to our application without exposing it to the stack configuration, like on GitHub.

- **Storage orchestration**

With Kubernetes and its plugins, we can automatically mount local and external storage solutions to the containers in a seamless manner, based on Software Defined Storage (SDS).

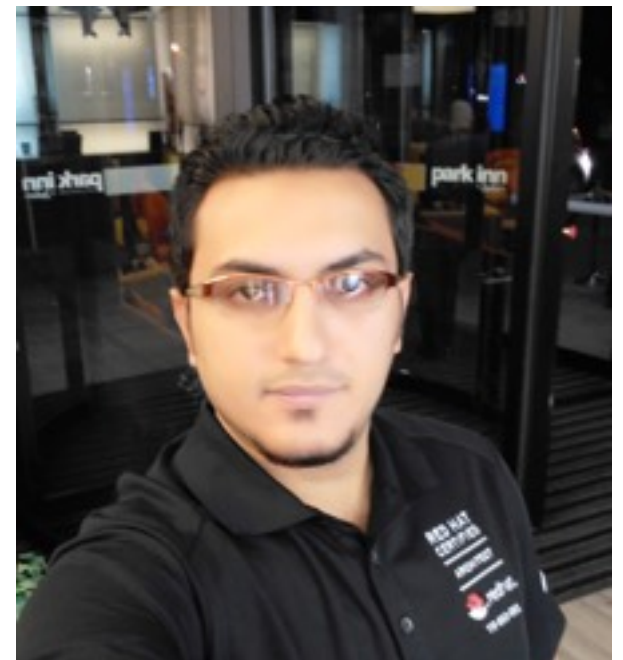
- **Batch execution**

Besides long running jobs, Kubernetes also supports batch execution.

There are many other features besides the ones we just mentioned, and they are currently in alpha/beta phase. They will add great value to any Kubernetes deployment once they become GA (generally available) features.

Meet the Author

Moustafa Nabil El-Zeny is a Principal UNIX/Linux and Open-Source and Security independent consultant with a huge profile of dealing and providing IT professional services, training, and consultation. He is one of the few certified RHCA all over the globe and one of only a few EMEA Instructors/Examiners (RHCI/RHCX) authorized to deliver both basic and advanced RH courses and exams. He masters all of Linux and UNIX family OSes.



He has been working as a Senior Red Hat Consultant, Solutions Architect for more than two years. He is senior UNIX/Linux Service Engineer and Solutions Specialist with 7+ years experience in UNIX and Linux industries. He is a Red Hat and Open-Source developer since 2005.

He has received close to 23 recognized international certificates and accreditations from Red Hat, ORACLE and edX.

He has successfully delivered a number of projects around GCC, MENA, and South Africa for more than 5 well-known and reputable international profiles such as Riyadh Bank - KSA, ADIP - Abu Dhabi, Government of Electricity - Dubai, Zain - Sudan, AWS - Cape Town, Etisalat Emirates - Dubai, SITA, SITA - KSA, Ministry Of Interior - KSA, Arab Bank - Amman, Bank Audi, Egypt with more enthusiasm and professionalism.

He has also conducted Red Hat Exam rounds for more than 500 people, tens of them were able to pass different Red Hat exams with impeccable scores. He is eager to share his knowledge with others and strives to be a successful resource helping people to migrate from proprietary systems to the freely open-source era!

Since the start of his career path in 2005, he was attached to one of his favorite songs lyrics belonging to R. Kelly "I believe I can fly, I believe I can touch the sky", because he achieved my dream. To reach the author, please contact him on LinkedIn:

<https://nl.linkedin.com/in/rhcamoustafaelzeny>

BSD Certification

The BSD Certification Group Inc. (BSDCG) is a non-profit organization committed to creating and maintaining a global certification standard for system administration on BSD based operating systems.

? **WHAT CERTIFICATIONS ARE AVAILABLE?**

BSDA: Entry-level certification suited for candidates with a general Unix background and at least six months of experience with BSD systems.

BSDP: Advanced certification for senior system administrators with at least three years of experience on BSD systems. Successful BSDP candidates are able to demonstrate strong to expert skills in BSD Unix system administration.

✓ **WHERE CAN I GET CERTIFIED?**

We're pleased to announce that after 7 months of negotiations and the work required to make the exam available in a computer based format, that the BSDA exam is now available at several hundred testing centers around the world. Paper based BSDA exams cost \$75 USD. Computer based BSDA exams cost \$150 USD. The price of the BSDP exams are yet to be determined.

Payments are made through our registration website:
<https://register.bsdcertification.org//register/payment>

i **WHERE CAN I GET MORE INFORMATION?**

More information and links to our mailing lists, LinkedIn groups, and Facebook group are available at our website:
<http://www.bsdcertification.org>

Registration for upcoming exam events is available at our registration website:
<https://register.bsdcertification.org//register/get-a-bsdcg-id>

How to Add a New System Tunable to FreeBSD

FreeBSD comes with several system tunables out of the box for each of its subsystems - there are tunables for virtual memory, file systems, I/O, networking, etc. We will learn how to customize them and also create our own system tunable.

What you will learn...

- Compile and install a custom FreeBSD kernel.
- Create a new system tunable.

What you should have...

- Familiarity with the C programming language.
- Command line familiarity

What you will need...

- A FreeBSD 11 installation

Installing FreeBSD kernel sources

If you did not install kernel sources when you installed FreeBSD, you can fetch the source

code - there are a couple of ways to obtain the kernel sources.

Using subversion to download the FreeBSD kernel sources

As root, install subversion – and check out the kernel sources with the following commands:

```
# pkg install subversion -y

# svn co --trust-server-cert --non-interactive
https://svn0.us-east.freebsd.org/base/stable/11/ /usr/src
```

What is a system tunable?

A system tunable is a variable which affects the way the kernel works. There are around 500 system tunables in FreeBSD and these variables can be modified at runtime. Some tunables can also be modified without a system reboot.

A system tunable can be read or written using the sysctl command. For example, we can read all the available variables on the system like thus:

```
$ sysctl -a
```

In our case we will add a **vm.proc_swapout_max** system tunable which can then be read and written using the following command:

```
$ sysctl vm.proc_swapout_max
```

Our new system tunable

Our new system tunable is inspired by Brendan Gregg's Scale x12 talk:

"Long before Unix supported paging, it used process swapping. While this was ok with the PDP-11/20's 64kB address spaces, it does not work as well today when address spaces can easily be hundreds of GB."(<https://www.illumos.org/issues/6583>).

This patch will allow us to either limit process swapping or or disable it entirely, with a system-configurable setting (you could disable swapping in your system using the system tunable `vm.swap_enabled = 0`, but doing that would defeat our purpose).

```
vm.proc_swapout_max
```

This new VM tunable allows limiting the swap-out of entire processes to only processes whose resident size (in bytes) is equal to or less than a given value (the default is 64kB)

To accomplish that, we will peek into the vm subsystem - specifically the paging subroutines. To achieve the goal set for this system tunable, we will modify `/usr/src/sys/vm/vm_glue.c` - go to line 845 using your favorite editor and add the following

```
/* Long before Unix supported
paging, it used process swapping.

* While this was ok with the
PDP-11/20's 64kB address spaces, it
does not work as well today

* when address spaces can easily be
hundreds of GB.*/

static u_long proc_swapout_max =
65536 ;

SYSCTL_ULONG(_vm, OID_AUTO,
proc_swapout_max, CTLFLAG_RW,
&proc_swapout_max, 0,

"Allows to limit the
swapout of whole processes whose max
resident size (in bytes) is equal or
less than value");
```

This is how you create a new system tunable - by using the SYSCTL(9) interface to add a new MIB (Management Information Base) entry.

Since we are using an unsigned long to represent the number of bytes, our tunable should use the `SYSCTL_ULONG` call which has the following signature:

```
SYSCTL_ULONG(parent, nbr, name, access, ptr, val, descr);
```

parent: Which group our new system tunable will live in (for example: `vm`, `vfs`, `kern`, etc..)

nbr: an OID number, as this is a new tunable, we need to use `OID_AUTO`.

name: the name of our system tunable.

access: We will read from and write to this variable.

ptr: a pointer to the variable that will hold the value of interest.

val: an initial value for this system tunable. Notice that we already have assigned a value to it.

descr: an accurate description of the purpose of this tunable.

Now we need to put our new variable to work. Looking at line 987, you will see this code:

```
/*
 * If the pageout daemon didn't free enough
 * pages,
 *
 * or if this process is idle and the system
 * is
 *
 * configured to swap proactively, swap it
 * out.
 */

if ((action & VM_SWAP_NORMAL) ||

    ((action & VM_SWAP_IDLE) &&

        (minslptime >
swap_idle_threshold2))) {
```

and change it to

```
/*
 * If the pageout daemon didn't free enough
 * pages,
 *
 * or if this process is idle and the system
 * is
 *
 * configured to swap proactively, and the
 * process resident count
 *
 * is less than vm.proc_swapout_max swap it
 * out.
 */

if (((vm_space_resident_count(p->p_vm_space)
 * PAGE_SIZE)
    <= proc_swapout_max ) &&
    ((action & VM_SWAP_NORMAL) ||
    ((action & VM_SWAP_IDLE) &&
        (minslptime >
swap_idle_threshold2)))) {
```

We added a new condition to filter processes based on their resident set size

`(vm_space_resident_count(p->p_vm_space) * PAGE_SIZE)` if they are less or equal to our `proc_swapout_max` variable. That's it - pretty simple (for more in-depth information on `p_vm_space` check `/usr/src/sys/sys/proc.h`).

We are now ready to test our changes, so next, let's build and install our kernel.

Installing our new system tunable

In case you have never built a custom kernel before - section 8.4 from the FreeBSD handbook may come in handy.

As root, follow these steps (assuming your machine architecture is also `amd64`)

```
# cd /usr/src/sys/amd64/conf
# mkdir /root/kernels
# cp GENERIC /root/kernels/NEWSYSCTL
# ln -s /root/kernels/NEWSYSCTL
```


These steps will create a new kernel configuration based on the GENERIC kernel and save it to /root/kernels so it's not lost in case you update your source tree.

```
# cd /usr/src
# make -j 4 buildkernel KERNCONF=NEWSYSCTL
```

This builds the kernel using the NEWSYSCTL configuration. The -j flag means execute at maximum 4 jobs - if you have more CPU cores, increase this number to help make building the kernel faster.

If all went well, we should now be able to install the new kernel. Again, as root:

```
# cd /usr/src && make install kernel
KERNCONF=NEWSYSCTL
```

Reboot your machine after this completes.

Testing our new tunable

We should now be able to see our new variable, just type:

```
# sysctl -a vm.proc_swapout_max
```

If the variable is found - congratulations you have added a system tunable to FreeBSD!

To test it, we must make the system exhaust memory and start swapping out processes (if you have disabled swap using vm.swap_enabled tunable, this will not work).

To stress your system, you could use a little program like the following:

```
#include <stdio.h>

#include <stdlib.h>

int main(int argc, char** argv)

{

    if (argc < 2)

    {

        printf("Need number of megabytes to allocate\n");

        exit(-1);

    }

    long nbr = atoi(argv[1]);

    printf("allocating %d megabytes\n",nbr);

    for(;;)

        malloc(1048576 * nbr);

}
```

This program will take as a parameter the number of megabytes that it will allocate in an infinite loop, so choose a number that will allow you to see the evolution on how your processes are swapped out.

You could use top to interactively see how your processes are behaving. Type w to check how much swap space is used by each process - that is the metric you will need to watch out for this new tunable.

Conclusion

Creating a new system tunable is really straight-forward, the most difficult part is deciding where and why to create one and getting acquainted with the subsystem you are modifying. It's a really helpful skill to have, allowing you to start disabling parts of the system, for example if you hit a bug that is not currently fixed or you have a specific use case where a system tunable could come in handy. Having access to the Design and Implementation of the FreeBSD Operative System helps a lot, as well as looking at the source code – which is always invaluable.

References

[https://www.freebsd.org/cgi/man.cgi?sysctl\(3\)](https://www.freebsd.org/cgi/man.cgi?sysctl(3))

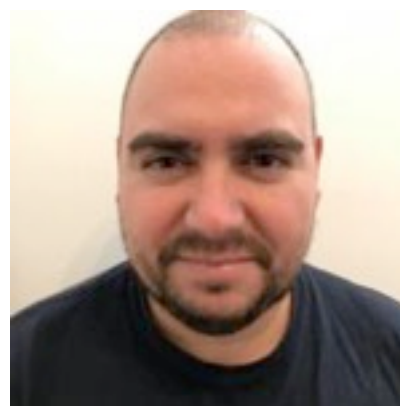
<https://www.freebsd.org/cgi/man.cgi?query=sysctl&sektion=9&manpath=FreeBSD+6.2-RELEASE>

https://svnweb.freebsd.org/base/releng/11.1/sys/vm/vm_glue.c?view=markup

<https://www.freebsd.org/doc/handbook/kernelconfig-building.html>

The Design and Implementation of the FreeBSD® Operating System, Second Edition

Meet the Author



Carlos Neira is a software engineer interested in performance, debugability and observability of systems. He has spent most of his career as a C and kernel programmer debugging issues on Linux, FreeBSD, Solaris and Z/OS environments. You can reach him at cneirabustos@gmail.com

HEY GOLIATH...



MEET DAVID

TRUENAS® PROVIDES MORE PERFORMANCE, FEATURES, AND CAPACITY PER-DOLLAR THAN ANY ENTERPRISE STORAGE ARRAY ON THE MARKET.

Introducing the TrueNAS X-Series: Perfectly suited for core-edge configurations and enterprise workloads such as backups, replication, and file sharing.

- ★ **Unified:** Simultaneous SAN, NAS, and object protocols to support multiple applications
- ★ **Scalable:** Up to 120 TB in 2U and 720 TB in 6U
- ★ **Fast:** Leverages flash and the Intel® Xeon® CPU with AES-NI for blazing performance
- ★ **Safe:** High Availability ensures business continuity and avoids downtime
- ★ **Reliable:** Uses OpenZFS to keep data safe
- ★ **Trusted:** TrueNAS is the Enterprise version of FreeNAS®, the world's #1 Open Source SDS
- ★ **Enterprise:** Enterprise-class storage including unlimited instant snapshots and advanced storage optimization at a lower cost than equivalent solutions from Dell EMC, NetApp, and others

The TrueNAS X10 and TrueNAS X20 represent a new class of enterprise storage. Get the full details at ixsystems.com/TrueNAS.



Caddy Web Server On FreeBSD

[What Is Caddy Web Server?](#)

[Caddy Features](#)

[How to Install Caddy in FreeBSD 11.1?](#)

[Caddy Configuration](#)

[Caddy Real Scenario](#)

What Is Caddy Web Server?

Caddy is an open source, middleware-enabled, secure, HTTP/2-enabled web server written in the Go programming language and started in 2015. Caddy configuration and initiation is so simple and clear – it allows you to create an HTTPS-enabled website in 5 seconds. In addition to this ease of use, the SSL certificate costs you nothing.

Caddy supports HTTP/2, and automatic TLS encryption. HTTP/2 is the HTTP protocol successor that can load websites faster.

Caddy automatically gets an SSL key and then serves your web site securely thanks to it's integration with Let'sEncrypt, a certificate authority which provides free TLS/SSL certificates.

Caddy supports a variety of Web technologies and is available as statically-compiled binaries for Windows, Mac, Linux, Android, and BSD

operating systems on i386, amd64, and ARM architectures.

A variety of web site technologies can be served with Caddy, which can also act as a reverse proxy and load balancer. Most of Caddy's features are implemented as middleware and exposed through directives in the Caddyfile (a text file used to configure Caddy).

Caddy is not vulnerable to a number of widespread CVEs including Heart-bleed, DROWN, POODLE, and BEAST. In addition, Caddy uses TLS_FALLBACK_SCSV to prevent protocol downgrade attacks.

Caddy Features

Notable Caddy features include:

HTTP/2 enabled

Server Name Indication (SNI)

OCSP (Online Certificate Status Protocol) Stapling

Virtual hosting

Native IPv4 and IPv6 support

Serve static files

Graceful restart/reload

Reverse proxy

Load balancing with health checks

FastCGI proxy

Templates

Markdown rendering

CGI via WebSockets

Gzip compression

Basic access authentication

URL rewriting

Redirects

File browsing

Access, error, and process logs

QUIC Support

How to Install Caddy in FreeBSD 11.1?

To install caddy, all you have to do is:

```
# pkg install caddy
```

You can simply issue “caddy -h” to get help on how to use caddy:

```
# caddy -h
```

-agree

Agree to the CA's Subscriber Agreement

-ca string

URL to certificate authority's ACME server directory (default "https://acme-v01.api.letsencrypt.org/directory")

-catimeout duration

Default ACME CA HTTP timeout

-conf string

Caddyfile to load (default "Caddyfile")

-cpu string

CPU cap (default "100%")

-disable-http-challenge

Disable the ACME HTTP challenge

-disable-tls-sni-challenge

Disable the ACME TLS-SNI challenge

-email string

Default ACME CA account email address

-grace duration

Maximum duration of graceful shutdown (default 5s)

-host string Default host

-http-port string

Default port to use for HTTP (default "80")

-http2

Use HTTP/2 (default true)

-https-port string

Default port to use for HTTPS (default "443")

-log string

Process log file

-pidfile string

Path to write pid file

-plugins

List installed plugins

-port string

Default port (default "2015")

-quic

Use experimental QUIC

-quiet

Quiet mode (no initialization output)

-revoke string

Hostname for which to revoke the certificate

-root string

Root path of default site (default ".")

-type string

Type of server to run (default "http")

-validate

Parse the Caddyfile but do not start the server

-version

Show version

Caddy Configuration

First, we create a directory and name it caddy:

```
# mkdir caddy
```

Then copy your index.html into it:

```
# cp index.html ./caddy/index.html
```

Next, go to this directory and issue the caddy command:

```
# caddy -host corebox.ir -cpu 50%  
-log log.txt -agree
```

Activating privacy features... done.

<https://corebox.ir>

<http://corebox.ir>

Then we can open “corebox.ir” in a browser. The point is caddy has automatically activated an SSL key.

A Real Scenario

In the real world we would need to restrict CPU cap, save web server logs or change the web server root directory.

In the next example we run our web server in the “/usr/local/www” directory. This command will cap CPU to 50 percent, save logs in “/var/log/caddy.log” and also agree to the CA's subscriber agreement.

```
# caddy -host corebox.ir -cpu 50%
-log "/var/log/caddy.log" -agree
-root "/usr/local/www".
```

You can create a file named Caddyfile and place all options into it:

```
# touch Caddyfile

# ee Caddyfile

corebox.ir

agree

browse

cpu 50%

log    /var/log/caddy.log
```

Caddy With API Access

In this example caddy proxies all API requests to a backend on port 9000.

```
# ee Caddyfile

corebox.ir

agree

browse

cpu 50%

log    /var/log/caddy.log

proxy  /api 127.0.0.1:9000
```

Conclusion

The Caddy web server is open source, but has features like QUIC which only enterprise web server supports and has a configuration syntax which is both clean and beautiful.

Useful Links

<https://github.com/mholt/caddy#quick-start>

<https://en.wikipedia.org/wiki/QUIC>

[https://en.wikipedia.org/wiki/Caddy_\(web_server\)](https://en.wikipedia.org/wiki/Caddy_(web_server))

<https://en.wikipedia.org/wiki/HTTP/2>

Meet the Author



Abdorrahman Homaei has been working as a software developer since 2000. He has used FreeBSD for more than ten years. He became involved with the meetBSD dot ir and performed serious training on FreeBSD. He is starting his own company (etesal amne sara tehran) in Feb 2017. his company is based in Iran's Silicon Valley.

Full CV: <http://in4bsd.com>

His company: <http://corebox.ir>

OpenBSD and The State of Gaming

OpenBSD is already well-known for its security strengths, but with its large collection of third party software, it can also be used for entertainment.

What you will learn...

- The extent of the possibilities of gaming
- The various existing repositories

What you need to know ...

- Some familiarity with OpenBSD's package installations
- In some cases, experience with compiling software from source (optional)

Indeed, more and more games have been ported over the years, from old to pretty recent ones. For instance, playing 3D games with relatively good performances is doable since OpenBSD supports very decent Intel chipsets.

Porting from other platforms

Most Open-Source games do not work directly on OpenBSD, at least originally. So the porting feasibility study is the first step. Luckily, it is doable most of the time. Usually, it is easier to port from FreeBSD rather than directly from Linux (but that does happen on some occasions), knowing the specifics of each platform can prove to be a great asset as often

the same sets of problems arise. Whenever possible, pushing those changes upstream (most of the time, it's a pretty modern repository either Github, Gitlab, Bitbucket, Subversion, but sometimes an "old fashioned" diff send by email to the author does the job too), at least the ones which make sense in a general multiplatform context, reducing the number of local patches accordingly. Pushing to the openbsd-wip repository is the second step before the port can possibly be accepted in the main port tree.

Available Games

We can always see the list of available playable games and engines in the port tree lists mentioned above. Most of the popular games for all tastes are happily introduced in the main cvs repository since enough releases (supertuxkart, supertux, chocolate-doom, 0ad just to name a few). However, there are other possibilities. If you're not against compiling the sources until they are at least under the openbsd-wip tree, where most of the games, even though all are not ready to be imported in the main tree, are in an acceptable state to be built and played. Thomas Frowhein (aka thfrw), an OpenBSD game port creator, edited this nice Gog.com list of available OpenBSD playable games.

https://www.gog.com/mix/openbsd_engine_available

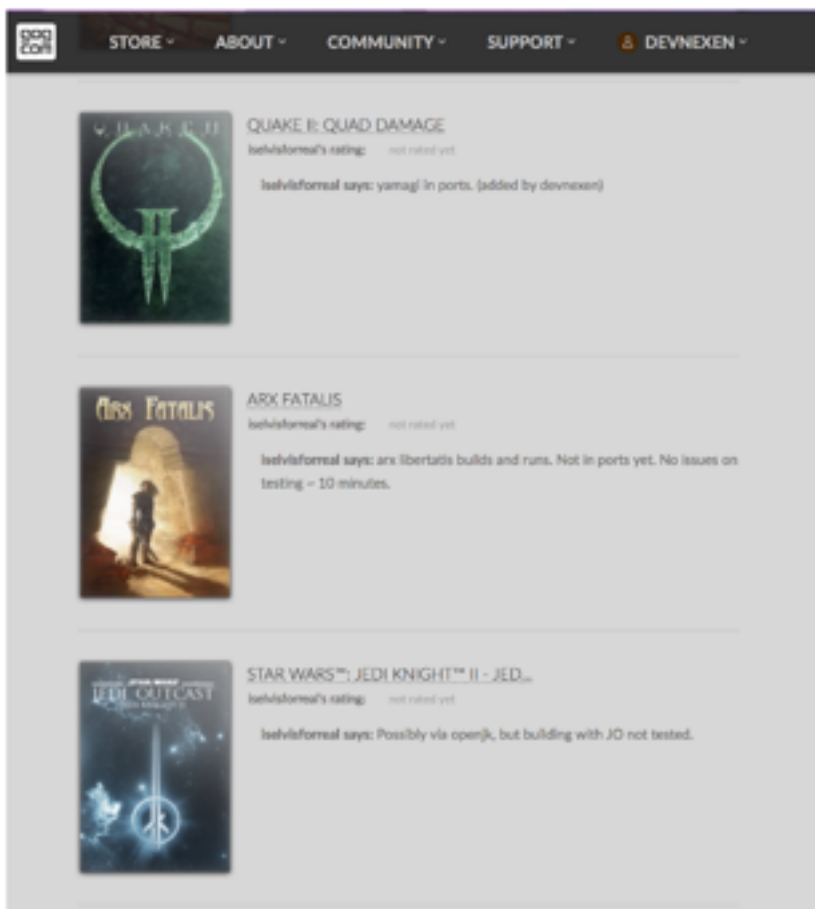


Figure 1. Gog.com page from thfrw

Recently, a certain amount of .NET/Mono games (FNA games to be more precise) had been tested by him and work seemingly well, but Mono would need a better support under OpenBSD. However, thfrw has been working on this for some time and might be able to fix it in a timely manner. Some significant recent additions like OpenJK, an engine for both Jedi Academy and Jedi Outcast, was added by Brian Callahan. Arx Libertatis for the popular Arx Fatalis and Barony, a 3D rogue game, can both be found on Gog and Steam. I singlehandedly ported them successfully, and surprisingly, created a potential of interest across all gamers irrespective of their ages since there is a limited number of such games.

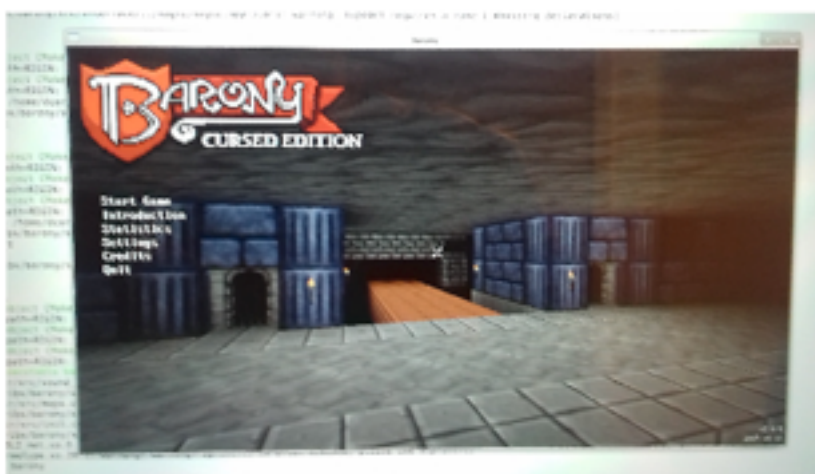


Figure 2. Barony, the popular 3D rogue game

Fs2open, a game engine for freespace 2.

Strife-ve, a doom based game. Also, OpenBSD has relatively good gamepad support.

Events

Adam Wolk, aka mulander, is a well-known OpenBSD contributor and hosts Quake I/Quake II/Quake III events. If you are interested, it is possible to know in advance when the events are scheduled.

https://www.reddit.com/r/openbsd_gaming/

Alternatively, you can join the #openbsd-gaming channel on Freenode to keep tabs on real-time information which is usually shared on Saturday evenings.

Conclusion

All of those are constantly “work in progress”, but OpenBSD has been proven to be a decent gaming platform. So if 2017 was a Desktop year, 2018 might be a Games year.

References

<https://github.com/openbsd/ports/tree/master/games> (main)

<https://github.com/jasperla/openbsd-wip/tree/master/games> (WIP)

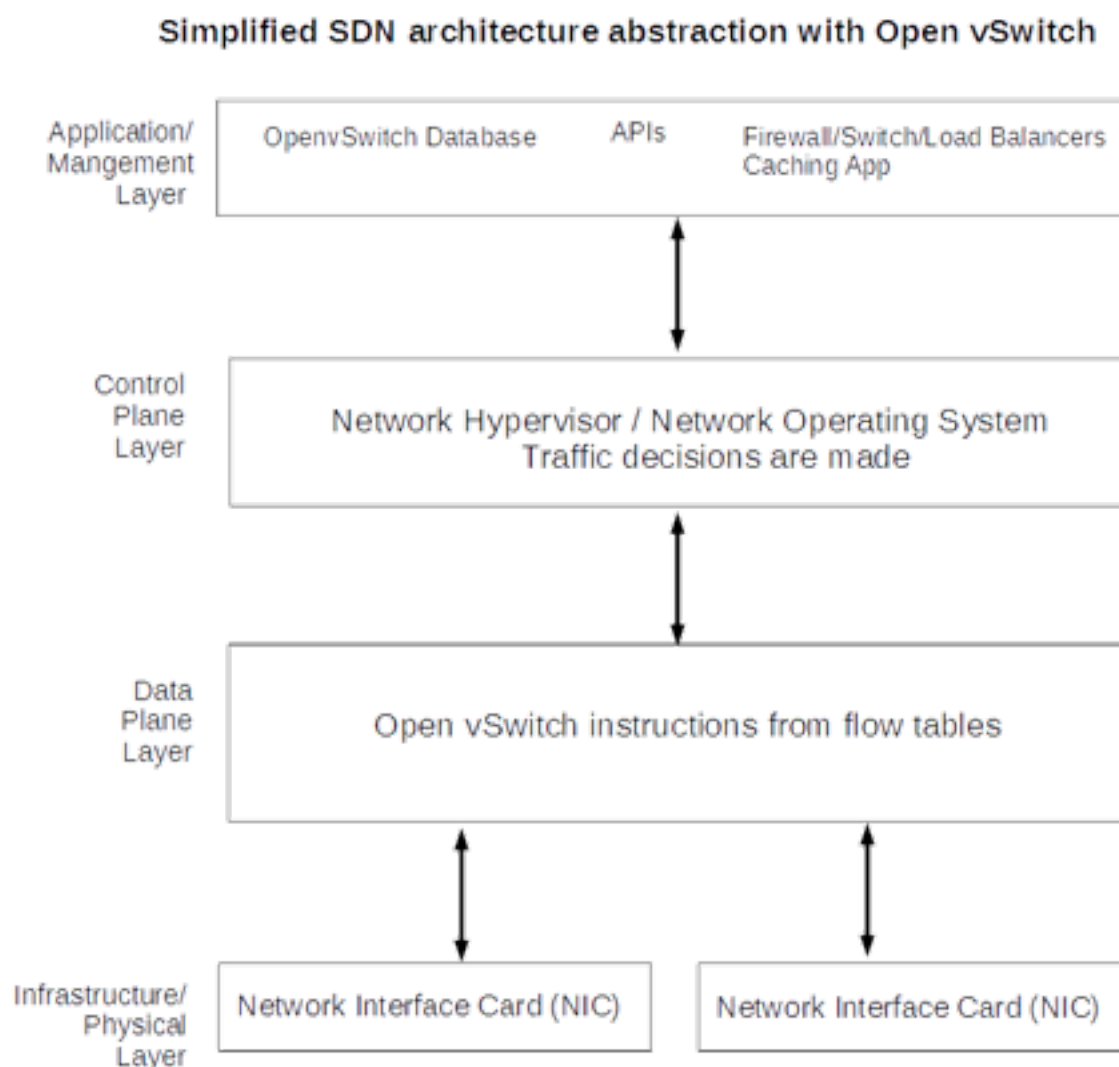
Meet the Author

David Carlier is a software developer since 2001, with several languages from C/C++ to Java, Python and Golang. He is working and living in Ireland since 2012's fall, co-organiser of Dublin BSD Group meetup.

Open vSwitch Overview

Open vSwitch (OVS) is an open source software defined networking solution to deliver software data center infrastructure as a service functionality for today's cloud based paradigms. OVS was built and based upon Stanford University's OpenFlow project. OVS functions both as a router and switch therefore is also referred to as a multilayer switch by examining content from the Open System Interconnection (OSI) reference model encompassing Layers 2 through Layer 7. OVS was designed for the dynamic and multi-server heterogeneous hypervisor virtualized environments for easy network stack management for virtualized infrastructure. OVS is supported the Linux, FreeBSD, NetBSD, Windows operating systems and has built default switch support for ESX, XenServer. Additionally, the data plane development kit (DPDK) provides a user level library interface this will be discussed in the later sections. We will now examine the key architectural features of the current stable release of OVS 2.9.0.

Open vSwitch Architecture



OVS components are comprised of OpenFlow and Open vSwitch Database. As you can see from the above diagram. Open vSwitch allows for elastic network configurations by managing packets as flows. A flow can be identified by any combinations of VLAN ID, Input port, Ethernet source/destination addresses, IP source/destination MAC addresses, TCP/UDP source and destination ports. Packets are sent to the controller and then the controller determines the action for the flow such as forward to port, ports, port mirroring, encapsulation forwarding to the controller or dropping the packet. The packet is then returned to the datapath or are handled by the data path.

Highlighted OVS Features

OVS contains a lot of supports a wide range of networking switch features and functions such as:

- native IPv4 and IPv6 addressing
- link aggregation (LACP IEEE 802.1AX-2008), Dot1q (802.1Q),
- NFV and VNF are management paradigms for controlling network services such as firewalling, NAT, DNS, caching and related services to be executed in software for consolidation
- virtual networking for open vswitch part of OVS 2.6
- Neutron integration networking-ovn openstack
- supports network ACLS distributed L3 routing for IPv4 and IPv6 – internal routing distributed on the hypervisor
- allow for ARP/ND suppression
- OVN: flow caching, decrement TTL
- built-in support for NAT, load balancing and DHCP services

- supports cloud technologies such as Kubernetes, Docker and Openstack

- features a built in DHCP server as part of the OVN agent

For further details, please consultant the link in the references section for additional details.

Software Defined Networking and Network Virtualization

Software Defined Networking (SDN) allows for the separation of the control plane and data plane. The control plane enables forwarding and routing switch decisions to be made. Similarly, the data plane allows for data forwarding to occur. The separation of control and data forwarding functionalities allows for network control to be programmable therefore allowing for forwarding layer abstraction to allow for easier portability to new hardware and software platforms.

Additionally, OVS functions as the point of egress for the overlay network which operate on top of physical networks within a data centre. OVS also allows for abstraction of network connectivity which been traditionally delivered via hardware for network virtualization. Network virtualization (NV) encompasses the virtualized L4 through L7 services, load balancing and firewalling applications. The ability to scale and adjust to the required resources demands meets the elastic requirements of cloud computing.

The data plane development kit (DPDK) is a bare metal cross-platform library and related drivers for fast user level hardware offloaded supported packet processing. It's designed to minimize the amount of CPU cycles required for fast sending and receiving functions. The performance gains achieved by using the DPDK interface is the result of bypassing the networking and kernel stacks. The DKDP was designed for use in specific network applications for network function virtualization (NFV) and enables mixed

Windows and Linux Kubernetes cluster orchestration.

An interesting feature of OVS is that it supports open virtual network (OVN) architecture is an abstraction for virtual networks. OVN allows OVS to function as a cloud management system for OpenStack integration and also can function as a gateway to allow for bi-directional traffic to be tunnelled in between physical Ethernet ports, this allows for transport mode functions to occur.

Open vSwitch Tutorial: KVM with OVS Bridge

The objective of this tutorial we will be using Open vSwitch on Ubuntu 16.04 64-bit and create an network bridge to connect the Linux KVM virtual machines.

1. Perform a new Ubuntu install (optional step)
2. Install Open vSwitch and the Linux Container and KVM package

```
$ sudo apt-get -y install openvswitch-switch  
qemu-kvm libvirt-bin ubuntu-vm-builder bridge-utils
```

3. Let's setup a KVM to use OVS as bridge

We verify the KVM install is good.

```
$ sudo virsh list --all
```

4. We will now create an OVS bridge which will be connected to KVM virtual machines running on. This will allow for KVM virtual machine to be associated with the internal OVS network.

NOTE: Please be careful when executing the next set of instructions as it may cause you to lose your connection if you're connected remotely to your server environment. It's recommended to play with open vswitch within a virtual machine testing environment.

We need to first disable Network Manager as Open vSwitch is not compatible with OVS switch. We will enable classic networking as the default.

We initialized the OVS database for initial startup

```
$ ovs-vsctl --no-wait init
```

Let's start open vSwitch daemon

```
$ sudo systemctl restart openvswitch-switch &&  
sudo systemctl enable openvswitch-switch
```

Let's create an Open vSwitch Bridge and verifying that the bridge has been created.

```
$ sudo ovs-vsctl add-br ovs-br0
```

```
$ sudo ip addr
```

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc  
noqueue state UNKNOWN group default qlen 1000
```

```
    link/loopback 00:00:00:00:00:00 brd  
    00:00:00:00:00:00
```

```
    inet 127.0.0.1/8 scope host lo
```

```
        valid_lft forever preferred_lft forever
```

```
    inet6 ::1/128 scope host
```

```
        valid_lft forever preferred_lft forever
```

...

```
4: ovs-br0: <BROADCAST,MULTICAST> mtu 1500  
qdisc noop state DOWN group default qlen 1000
```

```
    link/ether 9e:39:f8:46:eb:46 brd  
    ff:ff:ff:ff:ff:ff
```

We now display the created bridge interface properties.

```
$ sudo ovs-vsctl list bridge
```

```
_uuid          :  
46f8399e-9d46-46eb-b015-e0f80a4429cd  
  
auto_attach    : []  
  
controller     : []  
  
datapath_id    : "00009e39f846eb46"  
  
datapath_type  : ""  
  
datapath_version : "<unknown>"  
  
external_ids   : {}  
  
fail_mode      : []
```



```
flood_vlans      : []

flow_tables      : {}

ipfix             : []

mcast_snooping_enable: false

mirrors           : []

name              : "ovs-br0"

netflow           : []

other_config      : {}

ports             :
[915e6628-e720-439c-9e35-37bc8ad69fb6]

protocols         : []

rstp_enable       : false

rstp_status       : {}

sflow             : []

status            : {}

stp_enable        : false
```

5. We will now create a KVM network for OVS bridge and connected to KVM virtual machine

Let's create a new KVM network configuration:

```
cat <<EOF> ovs-network.xml

<network>

<name>ovs-bridgenet</name>

<forward mode='bridge'/>

<bridge name='ovs-br0'/>

<virtualport type='openvswitch'/>

</network>

EOF
```

We will enable libvirt network to be autostarted on host boot using the following commands:

```
$ sudo virsh net-define ovs-network.xml

Network ovs-bridgenet defined from
ovs-network.xml

$ sudo virsh net-start ovs-bridgenet

Network ovs-bridgenet started

$ sudo virsh net-autostart ovs-bridgenet

Network ovs-bridgenet marked as autostarted

$ sudo virsh net-info ovs-bridgenet

Name:                ovs-bridgenet

UUID:
e611f384-2e9a-4669-ac5f-447533edc3a0

Active:               yes

Persistent:           yes

Autostart:            yes

Bridge:               ovs-br0
```

6. We now will install VirtManager graphical interface for creating KVM virtual machines. For a local install we use the following commands:

```
$ sudo apt-get install -y virt-manager
```

For a remote install we need to install some additional packages:

```
$ sudo apt-get install -y virt-manager
ssh-askpass-gnome --no-install-recommends
```

```
$ sudo systemctl restart virtlockd.service &&
sudo systemctl enable virtlockd.service
```

```
$ sudo systemctl restart virtlockd.socket &&
sudo systemctl enable virtlockd.socket
```

```
$ sudo systemctl restart virtlogd.service &&
sudo systemctl enable virtlogd.service
```

```
$ sudo systemctl restart virtlogd.socket &&
sudo systemctl enable virtlogd.socket
```

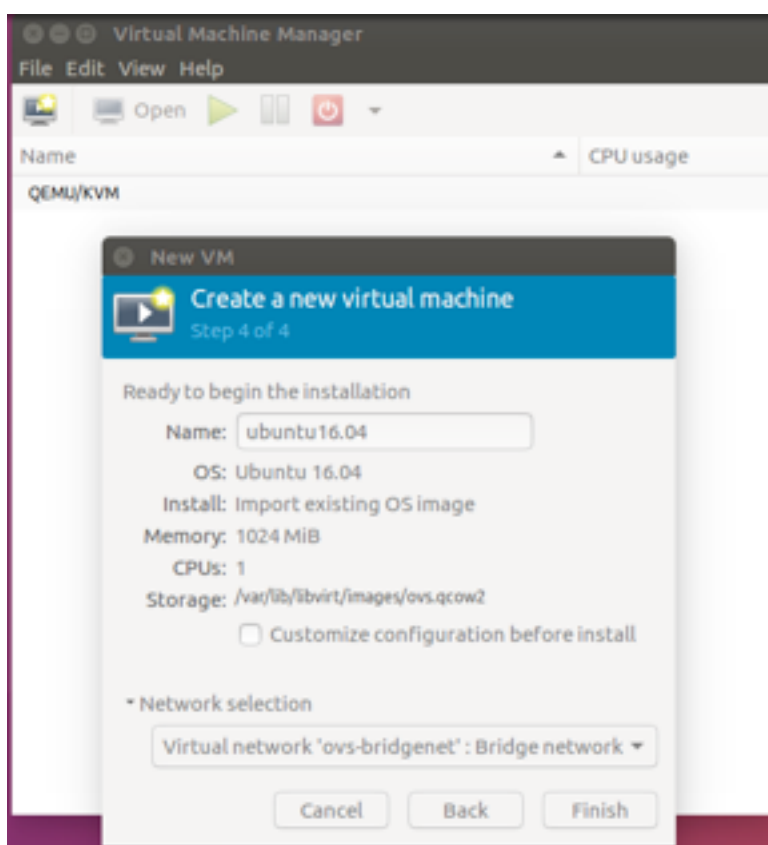
```
$ sudo usermod -a -G libvirtd sysop <replace
with your non root user>
```

7. We now launch virt-manager from Applications->System Tools -> Virtual Machine Manager or from the command line: `sudo virt-manager`. For demonstrative purposes we will use Ubuntu core for our KVM guest.

```
$ nice wget
http://cdimage.ubuntu.com/ubuntu-core/16/stable/current/ubuntu-core-16-amd64.img.xz
```

```
$ unxz ubuntu-core-16-amd64.img.xz
```

8. Create a new KVM VM and from the New Network of the new virtual machine creation wizard select ovs-bridgenet for the network selection as shown in the screen capture below.



9. Please select finish to complete the VM creation. The virtual machine will be launch and proceed to complete the guest VM install.

We now will setup static networking on the host and guest. For demonstrative purposes we will use the IPv4 address 10.0.0.1 with netmask 255.255.255.0 for the open vSwitch host using the command:

```
$ sudo ifconfig ovs-br0 10.0.0.1 netmask
255.255.255.0 up
```

For the KVM VM we will need to configure the network adaptor by using a similar command:

```
$ sudo ifconfig eth0 10.0.0.2 net mask
255.255.255.0 up
```

10. We can now test the connectivity between the host and the KVM VM via open vswitch by using the ping command to the guest.

```
$ sudo ping -c 5 10.0.0.2
```

```
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
```

```
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64
time=0.049 ms
```

```
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64
time=0.118 ms
```

```
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64
time=0.101 ms
```

```
64 bytes from 10.0.0.1: icmp_seq=4 ttl=64
time=0.121 ms
```

```
64 bytes from 10.0.0.1: icmp_seq=5 ttl=64
time=0.134 ms
```

```
--- 10.0.0.1 ping statistics ---
```

```
5 packets transmitted, 5 received, 0% packet
loss, time 4090ms
```

```
rtt min/avg/max/mdev = 0.049/0.104/0.134/0.031
ms
```

Conclusion

OVS is a versatile SDN framework which provides not only switch related functionality but supports various industry standard protocols and network features. The suite of development and related utilities provided by OVS is versatile tool for today's demanding cloud computing challenges.

References and Links

<https://www.ubuntu.com/containers/lxc>

<http://www.openvswitch.org/features/>

<http://www.openvswitch.org/support/dist-docs/ovn-architecture.7.html>

<http://www.openvswitch.org/support/dist-docs/>

<http://docs.openvswitch.org/en/latest/faq/issues/>

https://enigma.usenix.org/sites/default/files/nsdi15_full_proceedings_interior.pdf#page=125

<https://software.intel.com/en-us/articles/dpdk-performance-optimization-guidelines-white-paper>

https://access.redhat.com/documentation/en-us/reference_architectures/2017/html/deploying_mobile_networks_using_network_functions_virtualization/performance_and_optimization#figure16_caption

<http://www.openvswitch.org/support/boston2017/0900-ovn-on-windows.pdf>

<https://linuxcontainers.org/lxc/getting-started/>

Please refer to “Talk & Presentations” section for more conference talks

<http://www.openvswitch.org/>

Meet the Author

Albert Hui has been passionate about unix and other exotic operating systems and has been an OpenBSD enthusiast since 2003.

Presentation

How to Assist the Business World with OTRS?

Abstract

At Add-Ons for OTRS, we highly believe in the importance of any company to offer world-class *Customer Service*. As for today, customers have access to different technologies where they can qualify their customer experience with regard to a brand or enterprise. Therefore, we aim at highlighting the kindness of OTRS, an open-source software that is highly scalable and can be adjusted to address the most demanding requirements.

In this article you are going to find out:

- Why customer experience is key for business
- A wide selection of features available within OTRS
- OTRS installation requirements
- Features and installation process of Stop SLA for OTRS
- You might need experience in:
- Help Desk/Service Desk Software
- Open-Source software

OTRS

Introduction

In this article we aim to show OTRS open-source software from a business perspective. Therefore, the readers will have a deep insight on how important it is for any business to equip itself with a solution that can lift its performance.

We have started by highlighting the customer service experience perks to understand that customizable software is a key factor for enterprises when it comes to answering consumers inquiries or complaints.

Furthermore, we have described the key features of OTRS, an open-source software that tries to fit constantly to the business industry's demands by developing new attributes to its system and allowing companies to be the guide for its improvements.

Finally, we have touched the SLAs, SLAs are available as a one-time paid extension, as an important characteristic for customer service providers when solving their clients' concerns.

Why is the customer experience so important for a business?

Let's keep in mind that customer service is a part of *intangible marketing*. It provides companies with relevant information of current customers and gives service representative insight into the needs of potential ones. It guides businesses to detect opportunity areas, to develop and diversify their offer.

A grand customer service is the backbone of any business. Promotions and slash prices might serve as a customer magnet, but unless they can get some of those buyers to come back, the profitability of the business is not sustainable.

We all know this scenario from our own experience. We can intuit that grand customer service relies on offering the best possible experience to the clientele. Clients expect a quick, suitable and quality answer to their requests or complaints.

When companies focus on solving clients' inquiries, these can sense that their concerns are as important for their service provider as they are for them. As a result, firms successfully, turn them from happy customers into brand influencers.

Hence, companies of all sizes should be careful while choosing the appropriate platform to help them undertake this activity. *Because it can make a whole world of difference.*

Open-Source software

To face the customer service challenge, companies should equip their staff with a help desk solution that can simplify the work for the team. Open-source software enables businesses to work on long-term projects, modify, develop and customize them according to their needs.

Nowadays, B2C contact is done using different communication channels like calls, emails, chats

or social media. And at some point, it can just get confusing to track all the incoming inquiries.

Handling customer communication in a professional and efficient manner can be achieved by introducing OTRS to your company.

OTRS key features

OTRS is designed to provide companies with friendly software that will help them manage customer service efficiently.

OTRS (Open Ticket Request System) is an open source and free of charge software and can be easily installed on different platforms such as AIX, Linux, Free BSD, Mac Os 10.x, Open BSD, Solaris, and Window.

The entire system is based on tickets. Every single entry is marked and receives a unique number forming a ticket. These tickets are delivered to different customizable queues, which are also assigned to customizable groups and roles. Such features grant managers control over a vast list of tickets waiting to be solved.

OTRS key factors

Sophisticated ticket management

A powerful combination of tools that allow filtering, processing, escalating and resolving tickets, assigning priorities and responsibilities, managing users, their groups and roles.

ITIL/ITSM compliance

OTRS ITSM serves as an extension to the regular version of OTRS and deals with requirements and good practices included in the IT Infrastructure Library. It is based on solutions from the ITIL v3. ITIL is a library of recommendations which provides highly efficient services IT services with highest efficiency.

Multi-language support

As a fully multi-lingual system, OTRS supports more than 20 languages which makes it a perfect tool for non-English speaking environments.

Email interface

The sophisticated email interface allows OTRS to accept tickets over email, filters them into queues based on subject or recipient, and automate actions that depend on custom header lines. An auto-response system and an email templating interface can be used to create templates for typical customer problems. OTRS can also be configured to deliver email notifications of ticket changes using SMTP or Sendmail. The email interface also includes support for MIME, S/MIME and PGP.

OTRS Installation process

The installation process can be done in two ways, through pre-built binary packages or source code archive. Making the right choice of installation type depends on your needs. However, the second option allows you to edit and customize OTRS installation according to your needs.

It's worthy to highlight that to install the system, a web-server and a database are required.

Advanced Stop SLA for OTRS

As any other open-source solution, OTRS comes with numerous add-ons that make it easier to lift the service desk's team performance. A great deal of them come for free and are available to download on dedicated websites. Some however, which include highly custom features, are treated as premium add-ons. These modify your system in the most advanced way, giving agents the ability to handle their tasks more effectively and at hand, unlike the regular features offered in a non-customized system.

In this section you will learn how to install and set an add-on that gives the ability to incorporate a simpler and more practical manner to stop escalation time of a ticket. *Advanced Stop SLA*, which, customizes *Stop SLA* based on *Generic Agent*, manually stops the scale of any ticket with a dedicated button and adds a widget in *AgentTicketZoom* view to display any *Stop SLA* activities.

Such a practical tool helps resolve numerous problems service desk teams struggle with on a daily basis. For instance, guarding quality information is a key factor as time is key when solving tickets efficiently. Keep in mind that solving a ticket accurately might avoid repeat tickets and will leave us with good practices to be implemented.

Advanced Stop SLA for OTRS

To help out OTRS users, Add-Ons for OTRS team has developed the Advanced Stop SLA add-on.

Module Description

Advanced Stop SLA was created as an extension to, the Stop SLA package. It allows stopping the escalation of time based on ticket states. Nonetheless, with Advanced Stop SLA the possibility to pause the escalation time is broader. A user can set specific conditions to lapse the escalation time, which are set according to ticket attributes, such as queues, states, dynamic fields etc.

Further, Advanced Stop SLA incorporates a dedicated button to manually stop the escalation, if needed. This manual stop functionality can be restricted to owners of tickets or to a specific group.

Supported Versions

5.x.x. and 6.x.x.

1. Settings

Manual **StopSLA** button for ticket owners:

First, create an escalated ticket and go to details view. Once the escalated ticket has been created the **StopSLA** button is going to be visible on ticket’s action bar. Make sure to create an escalated ticket (in queue that has set SLA time or Service + SLA).



Click on the **StopSLA** button to pause the escalation. When the button has paused changes to **ResumeSLA** - that indicates that Manual **StopSLA** process is applied. Now, on the **AgentTicketZoom** within **StopSLAHistory** widget, a new activity about **StopSLA** will have been registered.



Click the **ResumeSLA** button – the escalation time will resume. The resuming action will be saved as well in the **StopSLAHistory** widget.

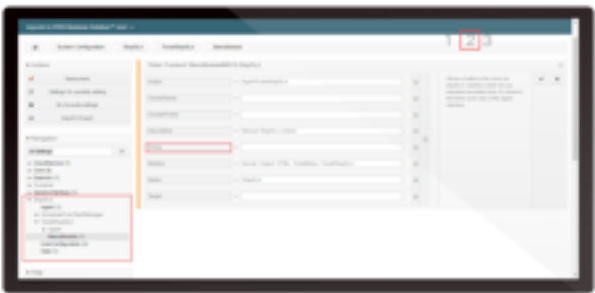


Manual **StopSLA button** for specific group only:

Go to **Admin**, locate **System Administration** and select **SysConfig Module**.



Search for, `StopSLA>TicketStopSLA>MenuModule`. Then select, the subgroup `StopSLA::TicketStopSLA::MenuModule`.



Locate the **Group** section and input the desired permission and group restriction in the order:

`permission:group;permission:group2;permission:groupN`

for example:

`rw:StopSLA-group1;rw:StopSLA-group2;rw:StopSLA-groupN;`



Each pair of permission and group should be divided by ';':

Automatic **StopSLA**

The Automatic **StopSLA** is a process that stops escalation time of a ticket automatically, based on **Generic Agent Module**. For example, pauses of time can be done at chosen state, when tickets obtain specific dynamic field, or when tickets are assigned to a specific queue etc.

Settings for **Automatic StopSLA**:

Go to **Admin**. Then, locate **System Administration** and select **Generic Agent Module**.



On the list of **Generic Agent** jobs locate: **StopSLA Automatic conditions**. This is a predefined example job created when the package is installed. Click on it to edit the job properties.



In the **Job Settings** window set **Validity** to **No** for now. The important sections for now are **Select Tickets** and **Execute Custom Modules**.



*Section such as **Update/Add Ticket Attributes**, **Add Note**, **Execute Ticket Commands** work as default Generic Agent job and can be used, but they will not be covered in this article.

***Automatic execution (multiple tickets)** and **Event based execution (single ticket)** should not be set as they will make the **GA** job run more times that it is supposed to.

Now, expand the **Select Tickets** section and search for field **State**.

According to our example, the ticket should stop escalation if it's switched to **Paused** state.



*Keep in mind that setting for two conditions will make the ticket to fulfill both to match. If you wish to set two conditions you need to create two separate jobs (e.g. One for state field and another for dynamic field).

Now expand the **Execute Custom Module** section and make sure that the field **Module** has the following value

Kernel::Modules::StopSLA_GenericAgent.



***Important!** Generic agent job is not an Automatic StopSLA condition unless it has this **Custom module** set. If **Custom module** is not set the Generic agent job will not perform **StopSLA** actions.

Now we can set the **Validity** of the job to **Yes** and select **Submit** to save the changes.



Now, click **Run this task** button on the job list to see which tickets meet the condition to have SLA stopped.



After Run this task button has been clicked, a list of tickets will be displayed. It is possible to click the ticket number to move to the ticket details.

If everything is right, please select **Run this job** to execute the job.



*Running the job for the condition is necessary if you wish to apply **StopSLA** to old tickets.

Now let's create a new escalated ticket to meet the condition we have set previously.



The **StopSLA history** widget

In the Advanced **StopSLA** module a widget displaying StopSLA actions is included in the **AgentTicketZoomView**.



The widget is shown in form of a list, which shows the overall time of **StopSLA** and the history of **StopSLA** events. The events are shown from the latest one at the top and first ones at the bottom. Also, they are divided into three categories:

Red – Stop events – indicates when **SLA** time was stopped manually or automatically.

Green – Resume events – indicates when **StopSLA** was ended and **SLA** time has been resumed.

Blue – Information events – shows information on **StopSLA** status change from **Manual** → **Automatic**, and the automatic condition that made **Automatic StopSLA** possible.

The StopSLA actions in ticket history

StopSLA actions are recorded in the ticket history and shown within Action **StopSLA**.



Conclusions

The article successfully reached the objective of refreshing the readers with a topic that they might dominate but analyzed from a business point of view. We have shown that business industries are eager to meet user-friendly software to lend them a hand at performing their business as usual activities.

In today's competitive business environment, picking the most suitable tool for managing your customer interactions could be one of the most crucial business decisions you are ever going to make. Choosing a well-established, secure and business-driven solution will not only help you commit better to your customers, but also ease daily processes like prioritizing tasks for your staff. Thanks to its open-source nature, OTRS, like no other service desk software, offers so much freedom, scalability and flexibility. These factors contribute to it being so often chosen by

market leaders in different business sectors worldwide.

Sources

Author: María Polett Ramos - Marketing Specialist.

Contributor, Kamila - Gancarz Marketing Manager & UX writer.

Aleksandra Stefaniszak – Junior Graphic Designer

Add-Ons for OTRS:

<https://otrsaddons.com/products/advanced-stop-sla-add-on>

Linux Journal (05-10-2010) by Vikran Vaswani.

Visible body: Make Customer Smile in 7 Easy Steps with OTRS. Linux Journal:

<http://www.linuxjournal.com/content/make-customers-smile-7-easy-steps-otrs>

WE'RE Add-Ons FOR OTRS

EMPOWER
your **TEAM**
with a tool
to lift their
performance

ADD-ONS FOR OTRS

Marketplace for OTRS

WE HAVE

the expertise in:

- Premium Add-Ons for OTRS
- OTRS Update on demand
- OTRS Implementation
- Add-Ons development for OTRS
- OTRS support 24/7

We adjust to ad hoc requests and solve demanding tasks remotely. We cooperate with partners worldwide such as France, Germany, Greece, Macedonia, Poland, the Netherlands, the USA and Uruguay.

CONTACT US



+48 694 509 649



info@otrsaddons.com



www.otrsaddons.com



OPGK Rzeszow S.A.
ul. Geodetow 1,
35-328 Rzeszow,
Poland

ADD-ONS FOR OTRS

With the latest chemical attack in the UK that has critically injured two individuals and seriously injured a serving police officer, what are the geopolitical, media and technical implications of this latest outrage?

by Rob Somerville

The poisoning of Sergei and Yulia Skripal on the 4th of March in Salisbury will go down in the history books as one of the greatest pyrrhic victories in the history of spycraft, diplomatic relations and a well-documented “readme” of exactly how not to execute a political assassination. If Russia, and indeed Vladimir Putin is responsible for this criminal act, on the world stage at the very least, it places the effectiveness of the Russian state and secret services somewhere far below North Korea considering the recent fatal VX attack on Kim Jong-nam by the alleged perpetrator, Kim Jong-un. As anyone with a good grasp of history will realise, the arena of spies, diplomatic relationships and power is soaked in treachery, half-truths, propaganda, blood and double-dealings to the point that the mind spins and the phrase “The enemy of my enemy is my friend” becomes a common ethical currency.

Personally, I am yet to be convinced that the Russian state had a hand in this vicious crime. Despite the knee jerk reactions of our Prime Minister, and the almost instant coalescing of your local neighbourhood hawks that want to leverage any excuse to demonise Russia on the pretext for war, I applaud the French President, Emmanuel Macron, for summing up this whole incident in the spirit of Inspector Clouseau. “Fantasy politics” were his exact words, and I can think of no more soothing a balm to my personal embarrassment as a British citizen who has to suffer the implications of the recent words uttered by our Prime Minister, Foreign Secretary, and the baying wolves in our Parliament that

subscribe to a united front on the basis of a patriotic herd mentality. The leader of the opposition, Jeremy Corbyn, tried in vain to introduce some sanity into this whole colossal witch hunt, but to no avail. He had the temerity to ask for one thing that professional IT teams ask for in any disaster scenario.

Evidence.

Regardless of the outcome of this incident, there is one coincidence that refuses to go away. The impact of social media is having a major impact on the outcome of geopolitics, and politicians cannot get away with the control of the narrative in the same way prior to the cold war. One might subscribe this medium-term erosion down to democracy and human progress over the past half century, but the cherry on the cake has been the technological progress that has connected individuals to a knowledge base pretty much unavailable in the last episode where East West relations were at such a nadir. In 1962, apart from the popular press your average citizen had no access to academic research papers or historical fact than was available at their local library. Today, it is a different matter entirely, and the chemical composition of Novichok is available at the press of an enter key, be it with a degree of traceability or near total anonymity. Individuals are no longer wallflowers, and personal opinion is rife on the internet, no matter how banal or revelatory. On one level, that is the current debate surrounding “fake news” and the exact definition of what is and what it isn’t carries as much weight as the definition of “conspiracy theorist”. It is a political weapon, a play on words that relies on character assassination, innuendo, suggestion and the subtle libel that implies the author or publisher is a sandwich short of a picnic or has ulterior motives in mind. Which is very interesting taking into account the current scandal surrounding both Facebook and Cambridge Analytica and the outcome of the 2016 US elections. Big data played a major part in the outcome, as will the influencing of the court of public opinion when it comes down to the Skripal affair.

In 1962, the matter was pretty cut and dried. The USA installed some missiles in Turkey, too close to the border of the USSR for their comfort. The USSR retaliated, and installed missiles in Cuba. After a Mexican stand-off, both sides aged a few years and decided that détente was the best option, and rolled back their nuclear missile development. With President Putin’s recent announcement concerning their development of missiles that can circumvent the ABM defences of the USA, the balance of power has now been redressed, as the American ABM technology effectively neutered any Russian nuclear strike be it aggressive or defensive. The \$64 million question is simple – are we in the West facing a Russia with new found confidence that is wanting to resurrect a weary and worn Cold War strategy of intimidation and provocation, or are we falling into a trap?

So in reality, the balance of power has now shifted more than ever into the hands of the technologists, scientists and those who stand for and believe in truth, honesty, and a better future for mankind. Unlike in 1962, this current tragedy will be played out in the living rooms, bedrooms, mobile phones and tablets of millions of citizens worldwide. Or to put it another way, any politician or state taking such an irresponsible gamble better be willing to have their case peer reviewed not just in the court of public opinion, but via international and world opinion. We potentially have two nuclear superpowers head to head, and the world is war weary. The appetite for global conquest is waning, and unlike the first and second world wars our youth are too attached to the internet to entertain fighting battles for a privileged few that can happily exist in an air conditioned bunker somewhere while the rest of us make do with the dining room table and a few sheets.

And that is the danger of the latest development, if this does turn nasty, as Einstein said we will wage the next war with sticks and stones. What is needed is a popular uprising on the internet and beyond, demanding and fostering discussion, dialogue, agreement and consensus not war, attrition and austerity. I'm sure there are those reading this article that would suggest that I am a Communist apologist, a Russian stooge. Far from it. Too many wars have been based on propaganda and patriotism, and the ability to communicate with anyone via the internet now totally negates that particular lever of power. Whoever organised that attack on the 4th of March has bitten off far more than they can chew, no matter what side they are on. If they wanted to demonise Russia, they will have failed as the case will be subject to international law and the evidence, so far, is rather thin on the ground and they will look rather stupid. If it was the Russian state, all this will do is drive a further wedge between West East relations that will not benefit the Russians, China or Korea (or indeed the West) in the long term.

There are few winners in this game.

The only conclusion I can come to in this whole matter is that some evil third party has decided to stir the pot a bit. I can but hope and pray that saner heads prevail, that the peacemakers and the doves will get a chance to sort this out rather than those that choose to rattle sabres, and take advantage of an already politically unstable political environment. We already have enough issues with Brexit and the internecine warfare surrounding the election of President Trump to contend with.

Among clouds Performance and Reliability is **critical**

Download syslog-ng Premium Edition
product evaluation [here](#)

Attend to a free logging tech webinar [here](#)



BalaBit
IT Security

www.balabit.com

syslog-ng log server

The world's first High-Speed Reliable Logging™ technology

HIGH-SPEED RELIABLE LOGGING

- above 500 000 messages per second
- zero message loss due to the
Reliable Log Transfer Protocol™
- trusted log transfer and storage

MAGAZINE BSD

FOR NOVICE AND ADVANCED USERS

UNIX

—How TO OPTIMALLY USE THE RESOURCES
OF YOUR MACHINE TO MINIMIZE THE RUN
TIME OF YOUR PROGRAM—

MARK SITKOWSKI
C.ENG, M.J.E.E DESIGN SIMULATION SYSTEMS LTD

ISSN 1898-9144

MAGAZINE BSD

FOR NOVICE AND ADVANCED USERS

APPLICATION DEBUGGING AND TROUBLESHOOTING

WORKSHOP
BY
CARLOS NEIRA

MAGAZINE BSD

FOR NOVICE AND ADVANCED USERS

PRACTICAL PYTHON 3.6.

LEARN TO PROGRAM USING PYTHON

WORKSHOP
BY
PEDRO ARAUJO - RUI SILVA

MAGAZINE BSD

FOR NOVICE AND ADVANCED USERS

HOW TO BUILD A FreeBSD KERNEL MODULE FROM SCRATCH

WORKSHOP
BY
DAVID CARLIER